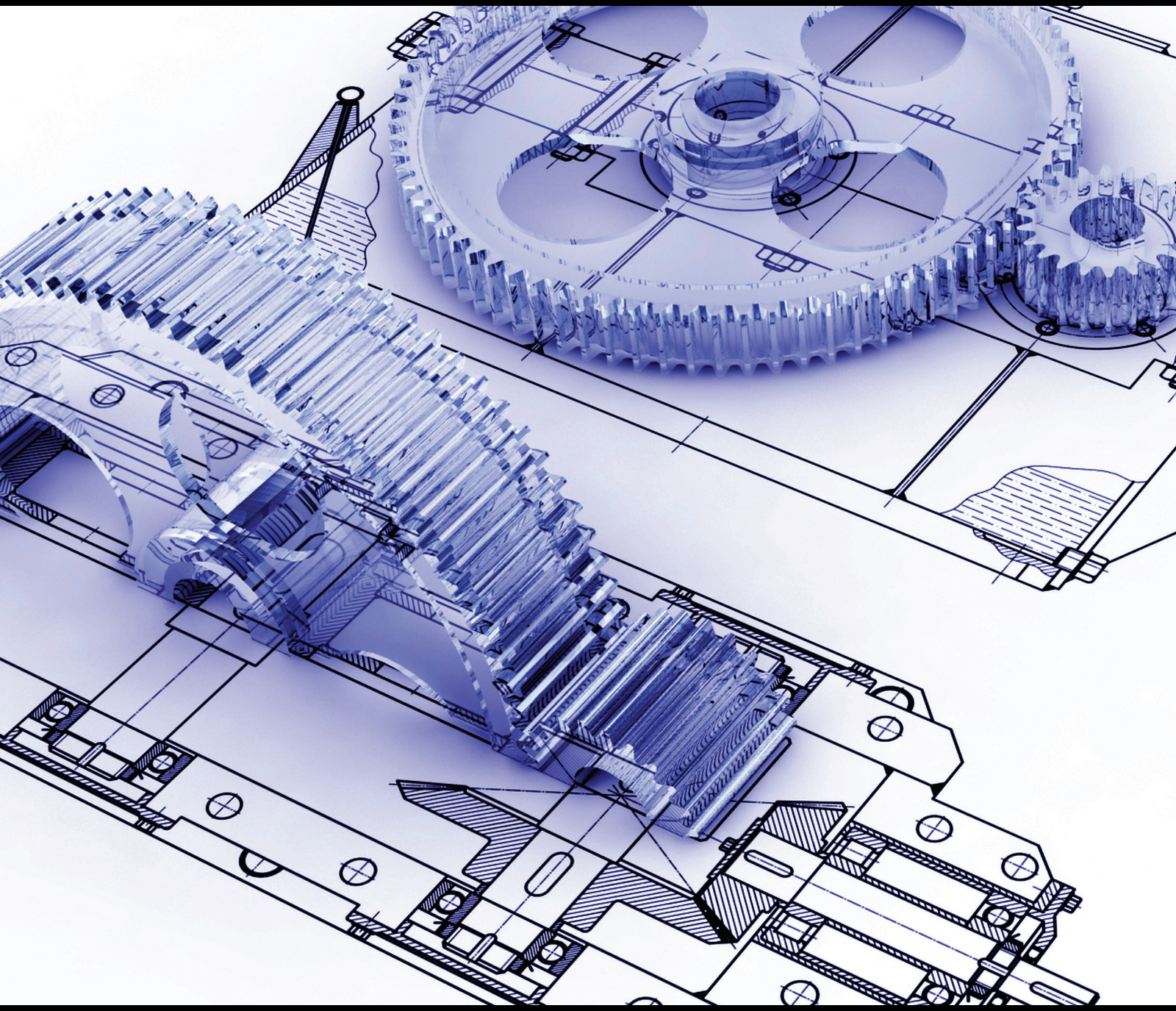
Advances in Mechanical Engineering

Advances of Artificial Intelligence in Mechanical Engineering

Guest Editors: Siamak Talatahari, Shengyong Chen, Amir H. Gandomi, and Amir H. Alavi



Advances of Artificial Intelligence in Mechanical Engineering

Advances in Mechanical Engineering

Advances of Artificial Intelligence in Mechanical Engineering

Guest Editors: Siamak Talatahari, Shengyong Chen, Amir H. Gandomi, and Amir H. Alavi

Editorial Board

Contents

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2014, Article ID 843730, 1 page
http://dx.doi.org/10.1155/2014/843730

Editorial

Advances of Artificial Intelligence in Mechanical Engineering

Siamak Talatahari,[1] Shengyong Chen,[2] Amir H. Gandomi,[3] and Amir H. Alavi[4]

[1] *Department of Civil Engineering, University of Tabriz, Tabriz, Iran*
[2] *School of Computer Science & Technology, Zhejiang University of Technology, Hangzhou, China*
[3] *Department of Civil Engineering, The University of Akron, Akron, OH 44325, USA*
[4] *Department of Civil and Environmental Engineering, Michigan State University, East Lansing, MI 48824, USA*

Correspondence should be addressed to Siamak Talatahari; siamak.talat@gmail.com

Received 22 July 2014; Accepted 22 July 2014; Published 12 August 2014

Artificial intelligence (AI) is a set of nature-inspired computational methodologies and approaches to address complex problems of the real-world applications to which traditional methodologies and approaches are ineffective or infeasible. AI methods and techniques, including neural networks, evolutionary computation, and fuzzy logic systems, have rapidly evolved over the last decades. Recently, the AI algorithms have attracted close attention of researchers and have also been applied successfully to solve problems in engineering. Nevertheless, for large and complex problems, the AI algorithms consume considerable computation time due to stochastic feature of the search approaches. Therefore, there is a potential requirement to develop efficient algorithm to find solutions under the limited resources, time, and money in real-world applications.

The papers selected for this special issue represent a good panel in recent challenges. The topics of the research papers and review papers are connected with the artificial intelligence methods and their application in mechanical engineering, for example, optimum design of mechanical systems; intelligent control, system identification, and damage detection; solid mechanics, geomechanics, material modeling, and smart materials; and structural dynamics, vibration, and inverse vibration problems.

Among many papers submitted to the special issue, all of the accepted papers were carefully reviewed and found appropriate for the journal; however clearly the topics and papers are not an exhaustive representation of the area of artificial intelligence in mechanical engineering. It can be seen that although some solutions and models become available, most problems remain open and research is highly active in this field. In the near future, we expect more contributions that will address all of the key aspects previously mentioned. Nonetheless, the special issue represents the recent concerns in the community and we have the pleasure of sharing them with the readers.

Acknowledgments

As the guest editors of the special issue, we would like to express our sincere appreciation to all the authors who contributed their work to this exciting special issue. We also thank the reviewers for their valuable and insightful comments that greatly benefited the improvement of papers quality.

Siamak Talatahari
Shengyong Chen
Amir H. Gandomi
Amir H. Alavi

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2014, Article ID 276264, 12 pages
http://dx.doi.org/10.1155/2014/276264

Research Article

Autonomous Robust Skill Generation Using Reinforcement Learning with Plant Variation

Kei Senda and Yurika Tani

Department of Aeronautics and Astronautics, Kyoto University, Nishikyo-ku, Kyoto 615-8540, Japan

Correspondence should be addressed to Kei Senda; senda@kuaero.kyoto-u.ac.jp

Received 1 December 2013; Revised 24 March 2014; Accepted 24 March 2014; Published 24 April 2014

Academic Editor: Amir H. Gandomi

This paper discusses an autonomous space robot for a truss structure assembly using some reinforcement learning. It is difficult for a space robot to complete contact tasks within a real environment, for example, a peg-in-hole task, because of error between the real environment and the controller model. In order to solve problems, we propose an autonomous space robot able to obtain proficient and robust skills by overcoming error to complete a task. The proposed approach develops skills by reinforcement learning that considers plant variation, that is, modeling error. Numerical simulations and experiments show the proposed method is useful in real environments.

1. Introduction

This study discusses an unresolved robotics issue: how to make a robot autonomous. A robot with manipulative skills capable of flexibly achieving tasks, like a human being, is desired. Autonomy is defined as "automation to achieve a task robustly." Skills can be considered to be solutions to achieve autonomy. Another aspect of a skill is including a solution method. Most human skill proficiency is acquired by experience. Since how to realize autonomy is not clear, skill development must include solution methods for unknown situations. Our problem is how to acquire skills autonomously, that is, how to robustly and automatically complete a task when the solution is unknown.

Reinforcement learning [1] is a promising solution, whereas direct applications of existing methods with reinforcement learning do not robustly complete tasks. Reinforcement learning is a framework in which a robot learns a policy or a control that optimizes an evaluation through trial and error. It is teacherless learning. By means of reinforcement learning, a robot develops an appropriate policy as mapping from state to action when an evaluation is given. The task objective is prescribed, but no specific action is taught. Reinforcement learning often needs many samples. The large number of samples is due to the large number of states and actions. So, online learning in a real environment is usually impractical. Most learning algorithms consist of two processes [2]: (1) online identification by trial and error sampling and (2) finding the optimal policy for the identified model. These two processes are not separated in typical learning algorithms such as Q-learning [3]. Reinforcement learning is said to be adaptive because it uses online identification and on-site optimal control design. Robustness attained using this adaptability is often impractical. It takes a very long time for online identification by means of trial and error.

In our approach, by learning a robust policy rather than by online identification, reinforcement learning is used to achieve a solution to an unknown task.

Using our approach, this study addresses an autonomous space robot for a truss structure assembly. It is difficult for a space robot to achieve a task, for example, peg-in-hole task, by contact with a real environment, because of the error between the real environment and the controller model. In order to solve the problem, a space robot must autonomously obtain proficiency and robust skills to counter the error in the model. Using the proposed approach, reinforcement learning can achieve a policy that is robust in the face of plant variation, that is, the modeling error. Numerical simulations and experiments show that a robust policy is effective in a real environment and the proposed method is used.

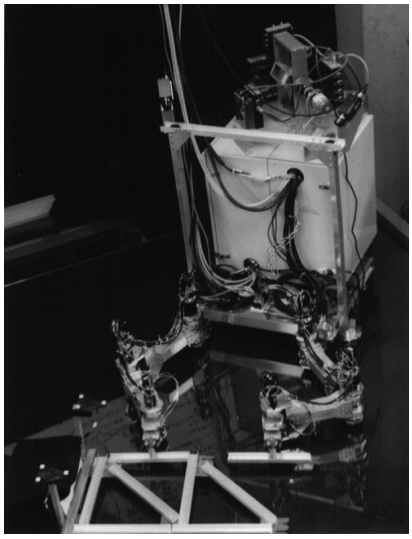


Figure 1: Photograph of space robot model and truss.

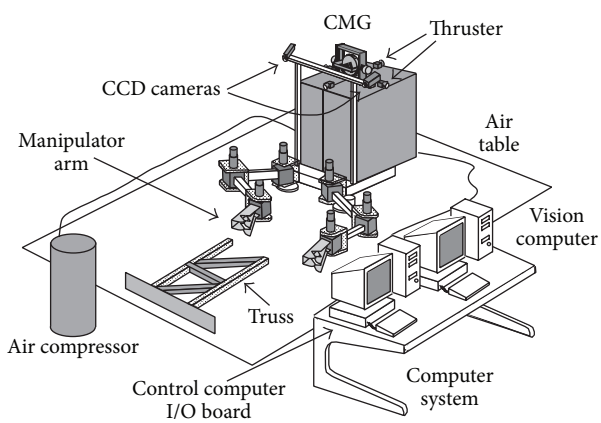


Figure 2: Schematic diagram of experimental system.

2. Need for Autonomy and Approach

2.1. Need for Autonomy. Autonomy is needed wherever robots work. Below, we discuss why and what kind of autonomy is required [4, 5] for space robots.

Space robots are required to complete tasks in the place of extra vehicular activity by an astronaut. Studies of autonomous systems are needed to realize space robots that can achieve missions under human-operator command. There are many applications for the autonomous space robots [6].

We developed ground experiments to simulate a free-flying space robot under orbital microgravity conditions (Figure 1). Using this apparatus, we have studied robot autonomy. Targeting control-based autonomy, we developed an automatic truss structure assembly, and so forth. However, it has been hard to achieve perfect robot automation because of various factors. To make a robot autonomous in the face of the following challenges, we must

(a) solve problems in the actual robot environment;

(b) operate robustly in the face of uncertainties and variations;

(c) overcome the difficulty of comprehensively predicting a wide variety of states;

(d) identify tasks and find unknown solutions to realize robust robot autonomy.

2.2. Approach to Autonomy Based on Reinforcement Learning. Human beings achieve tasks regardless of the above complicating factors, but robots cannot. Many discussions have rationalized the difficulties as originating in the nature of human skills. We have not established a means by which we can realize such skills. This study takes the following approach to this problem.

Section 4 approaches factors (a) and (b) in part by way of control-based automation taking account of the robustness of controls. The autonomous level of this approach is low because only small variations are allowable.

Section 5 considers how to surmount factors (a), (b), and (c) by learning. Using a predicted model and transferring the learned results to the actual system [7] is studied. This model is identified online and relearned. This procedure is applied where adaptability or policy modification is needed to thwart variations in the real environment. In some cases, the robot completes the targeted tasks autonomously. Learning is completed within an acceptable calculation time.

Section 6 considers factors (a), (b), (c), and (d). A peg-in-hole task has higher failure rates using the same approach as used in Section 5. Because of small differences between the model and the real environment, failure rates are excessive. The robot thus must gain skills autonomously. Skills similar to those of human beings must be generated autonomously. These skills are developed by reinforcement learning and additional procedures.

Section 7 evaluates approach robustness and control performance of the newly acquired skills. The skills obtained in Section 6 are better than those in Section 5.

3. Experimental System and Tasks

3.1. Outline of Experimental System. Our experimental system (Figures 1 and 2) simulates a free-flying space robot in orbit. Robot model movement is restricted to a two-dimensional plane [5].

The robot model consists of a satellite vehicle and dual three degrees of freedom (3-DOF) rigid SCARA manipulators. The satellite vehicle has CCD cameras for stereovision and a position/attitude control system. Each joint has a torque sensor and a servo controller for fine torque control of the output axis. Applied force and torque at the end-effector are calculated using measured joint torque. Air pads are used to support the space robot model on a frictionless planar table and to simulate the space environment. RTLinux is installed on a control computer to control the space robot model in real time. Stereo images from the two CCD cameras are captured by a video board and sent into an

image-processing computer with a Windows OS. The image-processing computer measures the position and orientation of target objects in the worksite by triangulation. Visual information is sent to the control computer via Ethernet.

The position and orientation measured by the stereo vision system involve errors caused by quantized images, lighting conditions at the worksite, and so forth. Time-averaged errors are almost constant in each measurement. Evaluated errors in the peg-in-hole experiments are modeled as described below. Hole position errors are modeled as a normal probability distribution, where the mean is $m = 0$ [mm] and standard deviation is $\sigma = 0.75$ [mm]. Hole orientation errors are modeled as a normal probability distribution, where the mean is $m = 0$ [rad] and standard deviation is $\sigma = 0.5\pi/180$ [rad].

We accomplish a hand-eye calibration to achieve tasks in the following sections. An end-effector, a manipulator hand, grasps a marker with a light-emitting diode (LED). The arm directs the marker to various locations. The robot calculates the marker location by using sensors mounted at the joints of the manipulator arm. The vision system also measures the marker location by using the stereo image by triangulation. Measurements using these joint-angle sensors have more precise resolution and accuracy. Hence, we calibrate the visual measurements based on measurements using the joint angle sensors. We consider the joint angle sensor measurement data to be the true value.

3.2. Tasks

3.2.1. Truss Assembly Task. Figure 3 illustrates the truss structure assembly sequence. The robot manipulates a truss component, connects it to a node, and proceeds each assembly step. Later this task is achieved by controls based upon mechanics understanding [5]. The truss design is robot friendly for easy assembling.

3.2.2. Peg-in-Hole Task. The peg-in-hole task is an example that is intrinsic to the nature of assembly. The peg-in-hole task involves interaction within the environment that is easily affected by uncertainties and variations, for example, errors in force applied by the robot, manufacturing accuracy, friction at contact points, and so forth. The peg easily transits to a state in which it can no longer move, for example, wedging or jamming [8]. Such variations cannot be modeled with required accuracy.

To complete a task in a given environment, a proposed method analyzes the human working process and applies the results to a robot [9]. Even if the human skill for a task can be analyzed, the results are not guaranteed to be applicable to a robot. Another method uses parameters in a force control designed by means of a simulation [10] but was not found to be effective in an environment with uncertainty. In yet another method [11], the task achievement ratios evaluated several predesigned paths in an environment with uncertainty. An optimal path is determined among the predesigned paths. There was the possibility a feasible solution did not exist among predesigned paths.

In the peg-in-hole experiment (Figure 4), the position and orientation of the hole are measured using a stereo camera. The robot manipulator inserts a square peg into a similar sized hole. This experiment is a two-dimensional plane problem (Figure 5). The space robot model coordinate system is defined as Σ_0, the end-effector coordinate system as Σ_E, and the hole coordinate system as Σ_{hl}. While the space robot completes its task, the robot grasps the structural site with another manipulator; the relative relation between Σ_0 and Σ_{hl} is fixed. State variables are defined as $[y_x, y_y, y_\theta, f_x, f_y, f_\theta]$, where (y_x, y_y) is the position of Σ_E in Σ_0, y_θ is the orientation about $\mathbf{k}_0$-axis, (f_x, f_y), and f_θ are the forces and torque in Σ_0 that end-effector applies to the environment.

The peg width is 74.0 [mm] and the hole width is 74.25 [mm]. The hole is only 0.25 [mm] wider than the peg. The positioning error is composed of the measurement error and the control error. The robot cannot insert the peg in the hole by position control if the positioning error is beyond ±0.125 [mm]. Just a single measurement error by stereo vision often moves the peg outside of the acceptable region.

4. Control-Based Automation

4.1. Truss Assembly Task. Automatic truss assembly was studied via control-based automation with mechanics understanding. The robot achieved an automatic truss structure assembly [5] by developing basic techniques and integrating them within the experimental system.

The following sensory feedback control [12] is used for controlling manipulators:

$$\boldsymbol{\tau} = -\mathbf{J}^T \mathbf{K}_P (\mathbf{y} - \mathbf{y}_d) - \mathbf{K}_D \dot{\mathbf{q}}, \tag{1}$$

where $\boldsymbol{\tau}$ is the control input to the manipulator and $\mathbf{J}$ is the Jacobean matrix. The $\mathbf{y}$ is the manipulation variable whose elements are the hand position/orientation $[y_x, y_y, y_\theta]$, $\mathbf{y}_d$ is the reference value of $\mathbf{y}$, $\mathbf{q}$ is the joint angle vector, and $\mathbf{K}_P$ and $\mathbf{K}_D$ are feedback gains. When the end-effector contacts the environment and manipulation variable $\mathbf{y}$ is stationary under constraint, the end-effector applies force and torque to the environment:

$$\mathbf{f} = -\mathbf{K}_P (\mathbf{y} - \mathbf{y}_d). \tag{2}$$

The force and torque can be controlled by $\mathbf{y}_d$. This is a compliant control.

Figure 3 is a series of photographs of the experimental assembly sequence. As shown in panel (i), the robot holds on to the worksite with its right arm to compensate for any reaction force during the assembly. The robot installs the first component, member 1, during panels (ii) and (iii). The robot installs other members successively and assembles one truss unit, panels (iv)–(vi).

Control procedures for the assembly sequence are as follows. There are target markers in the experiment environment as shown in Figures 1 and 3. Target markers are located at the base of the truss structure and at the storage site for structural parts. Each target marker has three

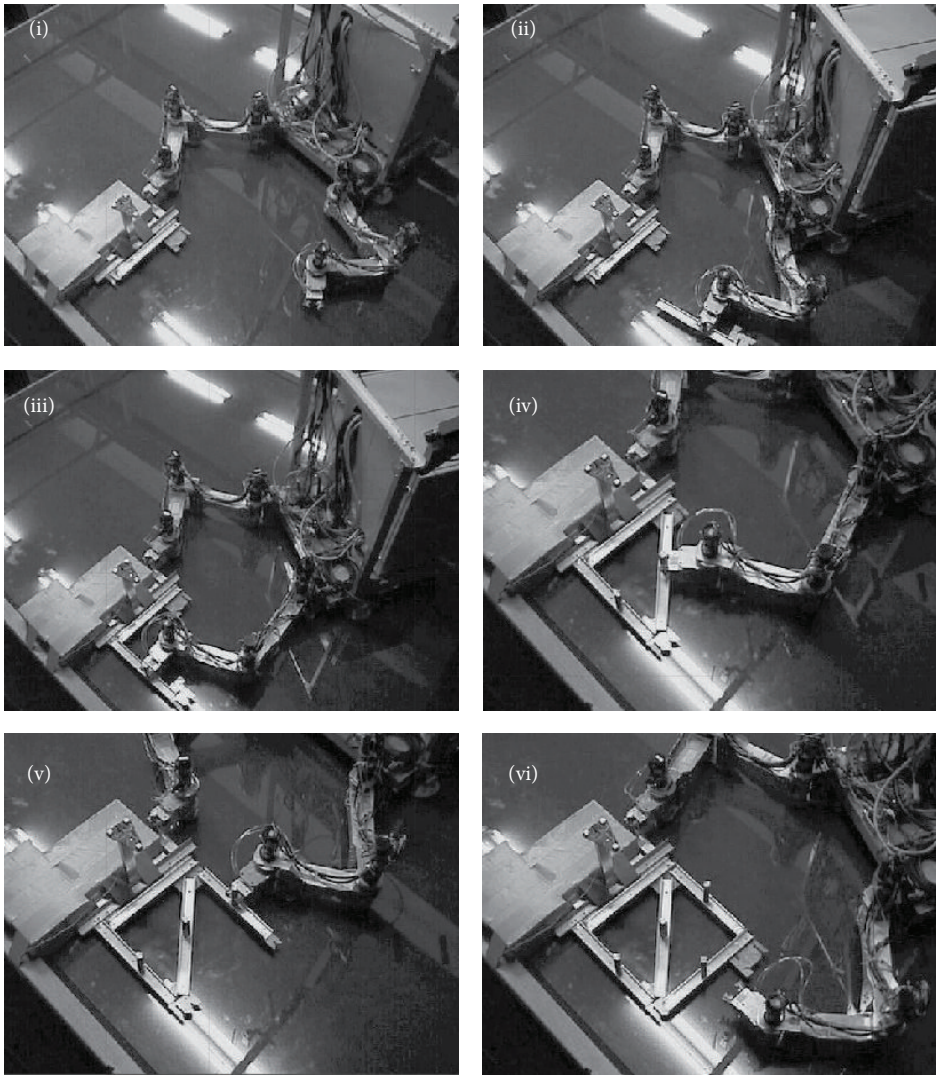


Figure 3: Sequence of truss structure assembly.

LEDs at triangular vertices. The vision system measures the marker position and orientation simultaneously. The robot recognizes the position of a part relative to the target marker before assembly. The robot places the end-effector position at the pick-up point, which is calculated from the target marker position as measured by the stereo vision system. At the pick-up point, the end-effector grasps a handgrip attached to the targeted part. The position and orientation of the part to the end-effector are settled uniquely when the end-effector grasps the handgrip. The robot plans the path of the arm and part to avoid collision with any other object in the work environment. It controls the arm to track along the planned path. The robot plans a path from the pick-up point to the placement point, avoiding obstacles by means of an artificial potential method [13]. Objects in the environment, for example, the truss under assembly, are regarded as obstacles. The arm is then directed along a planned trajectory by the sensory feedback control (1). The end-effector only makes contact with the environment when it picks up or places the part. Hence, feedback gains in (1) are chosen to make it a compliant control.

Consequently, the truss assembly task is successfully completed by control-based automation. However, measurement error in the vision system sensors, and so forth, prevents assembly from being guaranteed. Irrespective of uncertainties and variations at the worksite, the space robot model requires autonomy to complete the task goal.

4.2. Peg-in-Hole Task. The positioning control of (1) tries to complete the peg-in-hole task. The peg first is positioned at $y_\theta = 0$ [rad], it transits to the central axis of the hole, and it moves in a positive direction, toward $\mathbf{i}_0$. The peg does not contact the environment during transition from the initial state to the goal. Insertion is successful if the position control of the peg relative to the hole is free from error. Unfortunately, the peg-in-hole task is often unsuccessful because of the existing error. The robot cannot insert the peg in the hole by position control if the positioning error

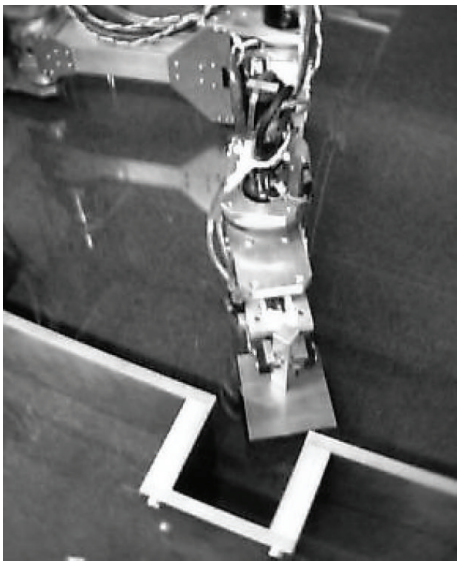


Figure 4: Photograph of peg-in-hole experiment setup.

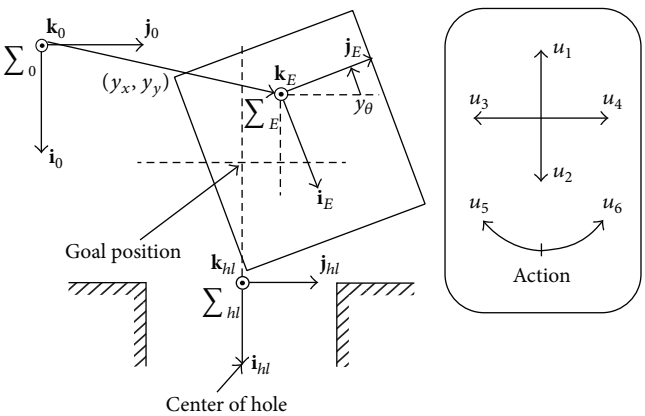


Figure 5: Definition of peg-in-hole task.

is greater than ±0.125 [mm]. So, single measurement error using stereo vision is often beyond the acceptable error. The manner in which the task fails is almost the same as shown in Figure 8 in the next section.

5. Existing Method for Autonomy with Reinforcement Learning

5.1. Outline of Existing Method with Reinforcement Learning. Reinforcement learning [1] is used to generate autonomous robot action.

In "standard" learning (Figure 6(a)), controller $\mathbf{K}_Q$ is designed in advance by learning the nominal plant model $\mathbf{P}_N$, and it is applied to real plant $\mathbf{P}$. We use a policy called controller $\mathbf{K}_Q$, which is designed with reinforcement learning methods. When variations exist, for example, measurement error in the vision system, unexpected obstacles appear in the environment, and so forth, and $\mathbf{K}_Q$ cannot complete tasks due to poor robustness and adaptability.

As shown in Figure 6(b), new plant model $\mathbf{P}'_N$ is reconstructed using visual measurement. Controller $\mathbf{K}'_Q$ is designed for the reconstructed model $\mathbf{P}'_N$. Controller $\mathbf{K}'_Q$ is then applied to real plant $\mathbf{P}$. Learning converges within a practical calculation time and the new policy is applicable to the truss structure assembly [5]. This method works well because it treats the kinematic problem without force interaction between the robot and the environment. The plant model for learning is reconstructed by visual measurement within a short time. This method has adaptability only if the model can be reconstructed accurately within a short time.

If the robot cannot complete the task with the controller due to error between the model and the real plant, the robot switches to online learning. Adaptability is realized by online identification and learning. However, this cannot be used for peg-in-hole task, because online identification requires too much time. In the next section, a reinforcement learning problem for a peg-in-hole task requires several tens of thousands of state-action pairs. It requires tens of days for online identification if a hundred samples are selected for each state-action pair and each sampling takes one second.

5.2. Existing Method with Reinforcement Learning

5.2.1. Problem Definition. Following general dynamic programming (DP) formulations, this paper treats a discrete-time dynamic system in a reinforcement learning problem. A state s_i and an action u_k are the discrete variables and the elements of finite sets $\mathcal{S}$ and $\mathcal{U}$, respectively. The state set $\mathcal{S}$ is composed of N_s states denoted by $s_1, s_2, \ldots, s_{N_s}$ and an additional termination state s_0. The action set $\mathcal{U}$ is composed of K actions denoted by $u_1, u_2, \ldots, u_K$. If an agent is in state s_i and chooses action u_k, it will move to state s_j and incur a one-step cost $g(s_i, u_k, s_j)$ within state transition probability $p_{ij}(u_k)$. This transition is denoted by (s_i, u_k, s_j). There is a cost-free termination state s_0, where $p_{00}(u_k) = 1$, $g(s_0, u_k, s_0) = 0$, and $Q(s_0, u_k) = 0$, $\forall u_k$. We assume that the state transition probability $p_{ij}(u_k)$ is dependent on only current state s_i and action u_k. This is called a discrete-time finite Markov decision process (MDP). The system does not explicitly depend on time. Stationary policy μ is a function mapping states into actions with $\mu(s_i) = u_k \in \mathcal{U}$, and μ is given by the corresponding time-independent action selection probability $\pi(s_i, u_k)$.

In this study, we deal with an infinite horizon problem where the cost accumulates indefinitely. The expected total cost starting from an initial state $s^0 = s_i$ at time $t = 0$ and using a stationary policy μ is

$$J^{\mu}(s_i) = \underset{s^1, s^2, \ldots}{E}\left[\sum_{t=0}^{\infty} g\left(s^t, \mu\left(s^t\right), s^{t+1}\right) \mid s^0 = s_i\right], \quad (3)$$

where $E_x[\cdot]$ denotes an expected value, and this cost is called J-factor. Because of the Markov property, a J-factor of

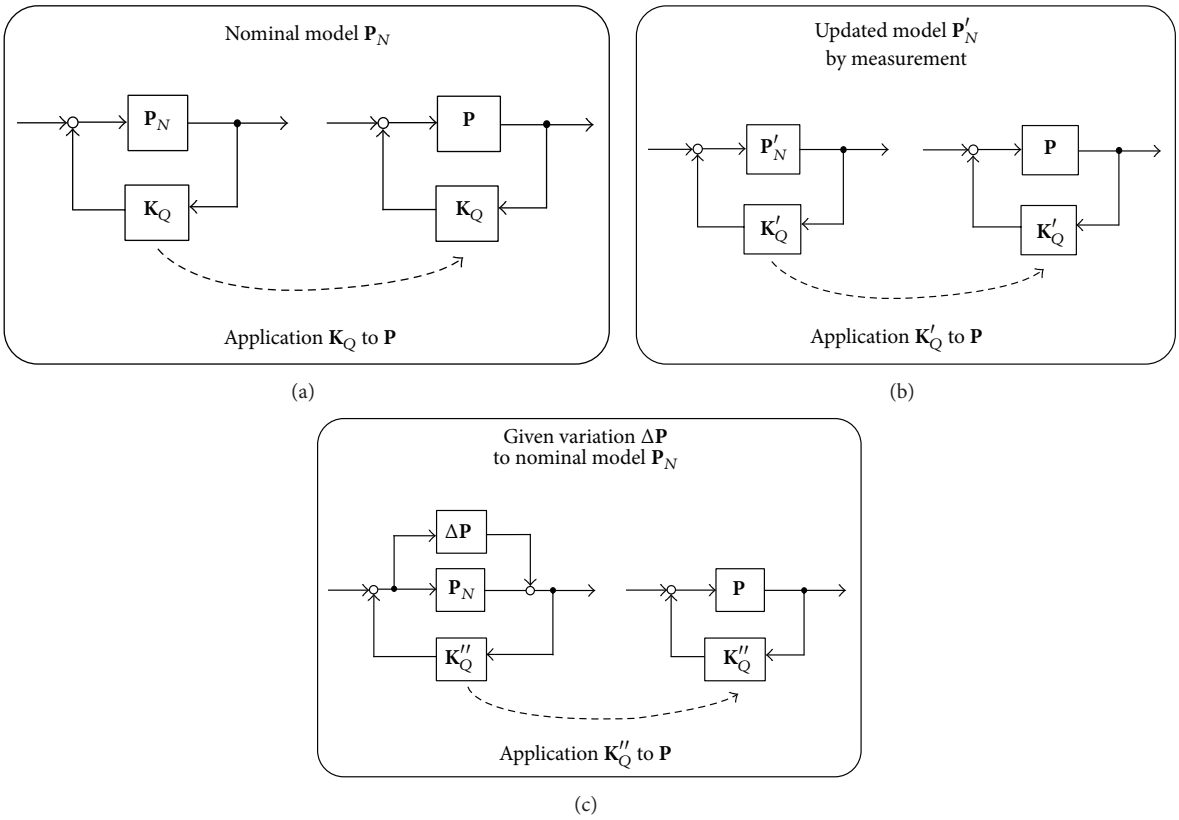


Figure 6: Learning using (a) nominal plant model, (b) updated plant model, and (c) plant model with variation.

a policy μ satisfies

$$J^{\mu}(s_i) = \sum_{k=1}^{K} \pi(s_i, u_k) \sum_{j=0}^{N_s} p_{ij}(u_k) \left\{ g(s_i, u_k, s_j) + J^{\mu}(s_j) \right\}, \quad \forall s_i. \tag{4}$$

A policy μ is said to be proper if μ satisfies $J^{\mu}(s_i) < \infty, \forall s_i$.

We regard the J-factor of every state as an evaluation value, and the optimal policy μ^* is defined as the policy that minimizes the J-factor:

$$\mu^*(s_i) \equiv \arg\min_{\mu} \sum_{i=1}^{N_s} J^{\mu}(s_i), \quad \forall s_i. \tag{5}$$

The J-factor of the optimal policy is defined as the optimal J-factor. It is denoted by $J^*(s_i)$.

The optimal policy defined by (5) satisfies Bellman's principle of optimality. Then, the optimal policy is stationary and deterministic. The optimal policy can be solved by minimizing the J-factor of each state independently. Hence, the optimal J-factors satisfy the following Bellman equation, and the optimal policy is derived from the optimal J-factors:

$$J^*(s_i) = \min_{u_k} \sum_{j=0}^{N_s} p_{ij}(u_k) \left\{ g(s_i, u_k, s_j) + J^*(s_j) \right\}, \tag{6}$$

$$\mu^*(s_i) = \arg\min_{u_k} \sum_{j=0}^{N_s} p_{ij}(u_k) \left\{ g(s_i, u_k, s_j) + J^*(s_j) \right\}. \tag{7}$$

5.2.2. Solutions. The existing type of reinforcement learning problem is solved as "standard" learning in Figure 6(a). It obtains the optimal policy μ^*_{nom}, which minimizes the J-factors of (7) for the nominal plant. It corresponds to controller $\mathbf{K}_Q$ in Figure 6(a). The optimal J-factor J^*_{nom} of μ^*_{nom} can be obtained by the DP-based solutions. The solutions are mentioned in [1, 2], but they are omitted here.

5.3. Learning Skill for Peg-in-Hole by Existing Method

5.3.1. Problem Definition of Peg-in-Hole. Here, the peg-in-hole task defined in Section 3.2.2 is redefined as a reinforcement learning problem.

State variables $[y_x, y_y, y_\theta, f_x, f_y, f_\theta]$ in Section 3.2.2 are continuous but discretized into 1.0 [mm], 1.0 [mm], $0.5\pi/180$ [rad], 2.0 [N], 1.0 [N], and 0.6 [Nm] in the model for reinforcement learning. The discrete state space has 4,500 discrete states, where the number of each state variable is

[5, 5, 5, 4, 3, 3]. Robot action at the end-effector is u_1, u_2, u_3, and u_4, at each of the end-effector states transiting by ± 1 in the direction of the $\mathbf{i}_0$-axis or $\mathbf{j}_0$-axis, and u_5 and u_6, at each of the end-effector states transiting by ± 1 about the $\mathbf{k}_0$-axis of rotation. State-action space is described in the space robot model coordinate system Σ_0. The hole is 0.25 [mm] wider than the peg, and (y_x, y_y) are quantized larger than this difference.

Control in (1) is used to transit from present state s_i to the next state s_j by action u_k. The reference manipulation variable to make the transition to s_j is $\mathbf{y}_d^{(s_j)} = \mathbf{y}_d^{(s_i)} + \delta\mathbf{y}_d^{(s_i)}$ given by $\delta\mathbf{y}_d^{(s_i)}(u_k)$ $(k = 1, 2, \ldots, 6)$, where $\delta\mathbf{y}_d^{(s_i)}$ is kept constant during transition. When the end-effector contacts the environment and manipulation variable $\mathbf{y}$ is stationary under constraint, the end-effector applies force and torque to the environment $\mathbf{f} = -\mathbf{K}_P(\mathbf{y} - \mathbf{y}_d)$ as (2), where $\mathbf{y}_d$ is the reference manipulation variable. Force and torque are controlled by $\mathbf{y}_d$, which is changed by an action. This is a compliant control, a force control. Tasks in which the end-effector contacts the environment, for example, peg-in-hole, demand a control with compliance. Therefore, a compliant control is essential as a basic control.

The robot takes the next action after (1) control settles and the peg becomes stationary. Regardless of whether the peg is in contact with the environment, the robot waits for settling and proceeds to the next action. For this reason, state variables do not include velocity.

The goal is to achieve states with the largest y_x, the peg position in $\mathbf{i}_0$-direction, in the state space. The one-step cost is $g(s_i, u_k, s_j) = 1$ for all states other than the goal state. Hence, the J-factor is the expected step number from s_i to the goal.

5.3.2. Learning Method. State transition probabilities for the state-action space in the previous section are calculated with sample data. Sample data are calculated with a dynamic simulator in a spatially continuous state space. The dynamic simulator is constructed with an open-source library, the open dynamics engine (ODE) developed by Russell Smith. The numerical model of this simulator has continuous space, force, and time in contrast to the discretized models for reinforcement learning in Section 5.2. This discretized state transition model is regarded as plant $\mathbf{P}_N$, and the method in Figure 6(a) is applied. The optimal policy μ^*_{nom}, that is, controller $\mathbf{K}_Q$, is derived from the solution in Section 5.2. The optimal policy μ^*_{nom} is applied to the dynamic simulator with a continuous state-action space or the hardware experimental setup, the real plant $\mathbf{P}$ in Figure 6(a). This study does not deal with the online learning.

5.3.3. Learning Result. The result of a numerical simulation in which controller $\mathbf{K}_Q$ is applied to the environment with no position/orientation error is shown. The peg moves and arrives at the goal as shown in Figure 7. Peg positioning is first changed to $y_\theta = 0$ [rad]. After the peg transits to the hole central axis, it is moved in a positive direction toward $\mathbf{i}_0$. Then the peg is inserted into the hole. During the transition from the initial state to the goal, the peg does not make contact with the environment, and the end-effector applies force and torque, $[f_x, f_y, f_\theta] = [0, 0, 0]$.

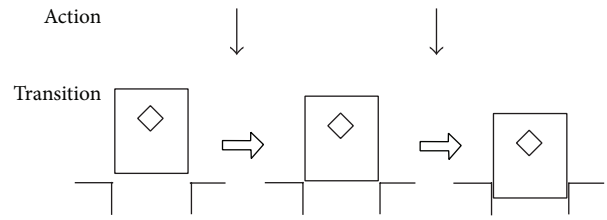


Figure 7: Trajectory of controller $\mathbf{K}_Q$ in a simulation without any hole position error.

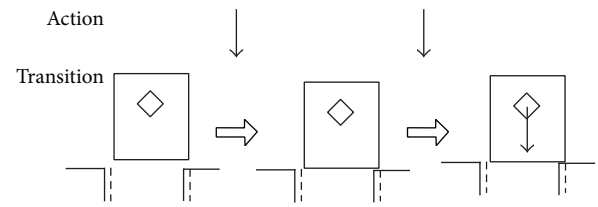


Figure 8: Trajectory of controller $\mathbf{K}_Q$ in a simulation with hole position error −0.5 [mm] in $\mathbf{j}_0$.

In an environment with the hole position error of −0.5 [mm] in $\mathbf{j}_0$ direction, the peg does not arrive at the goal with controller $\mathbf{K}_Q$; see Figure 8. The task is not completed using $\mathbf{K}_Q$ due to small errors caused by visual measurement, and so forth.

6. Autonomous Acquisition of Skill by Learning

6.1. Autonomous Acquisition of Skill for Peg-in-Hole. A robot can achieve peg positioning or movement with contact force, and it must have basic control functions same as a human being. Human vision measurement and positioning control are not accurate enough. However, the rate of a human failure in the same task is not as high as that of a robot. One reason for this may be the skills a human being brings to the task.

A human being conducting peg-in-hole task uses a typical sequence of actions (Figure 9). First, the human being puts a corner of the peg inside the hole. The peg orientation is inclined. The peg is in contact with the environment. Two points of the peg, the bottom and a side, are in contact with the environment, as shown in the close-up in Figure 9. The human then rotates the peg and pushes it against the hole and maintains the two contact points. The two corners are then inserted into the hole. Finally, the human inserts the peg into the hole and completes the task.

Human vision measurement accuracy and positioning control accuracy are not high. A human presumably develops skill while manipulating this situation. We conducted an experiment to check whether robot learning in the same situation could achieve the task as well as a human.

This situation conceptually corresponds to Figure 6(c). Plant $\mathbf{P}_N + \Delta\mathbf{P}$ denotes a variation plant with error caused by visual measurement, and so forth. Variation plant set $\{\mathbf{P}_N +$

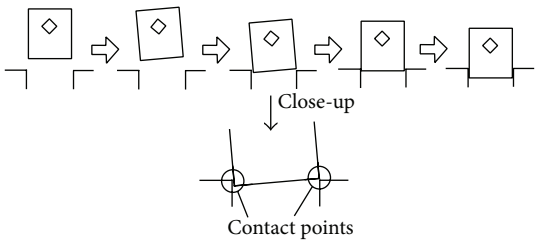


Figure 9: Human skill.

$\Delta\mathbf{P}\}$ is composed of all the variation plants that can exist. Real plant $\mathbf{P}$ is supposed to be a member of variation plant set $\{\mathbf{P}_N + \Delta\mathbf{P}\}$. The learning robot obtains controller $\mathbf{K}_Q''$. The controller is able to complete the task for all of the variation plants in $\{\mathbf{P}_N + \Delta\mathbf{P}\}$.

6.2. Problem Definition for Reinforcement Learning with Variation. We assume there are N variation plants around the estimated plant (the nominal plant). We use a set composed of N variation plants for learning.

We consider difference w_l between a variation plant and the nominal plant in each state, which is a discrete variable and the element of finite set $\mathscr{W}$. Finite set $\mathscr{W}$ is composed of L differences denoted by $w_0, w_1, \ldots, w_{L-1}$. Difference w_0 indicates no difference. If an agent is in state s_i with difference w_l and chooses action u_k, it will move to s_j within a state transition probability $p_{ij}(u_k; w_l)$ and incur a one-step cost $g(s_i, u_k, s_j; w_l)$. This transition is denoted by $(s_i, u_k, s_j; w_l)$. Difference w_l can be considered as the disturbance that causes state transition probability $p_{ij}(u_k; w_0)$ to vary to $p_{ij}(u_k; w_l)$. We assume that the $p_{ij}(u_k; w_l)$ and $g(s_i, u_k, s_j; w_l)$ are given.

Difference w_l at each state is determined by a variation plant. Variation η is a function mapping states into difference with $\eta(s_i) = w_l \in \mathscr{W}$. The nominal plant is defined by $\eta_0(s_i) = w_0$ for all states s_i. The plant does not explicitly depend on time, so variation η is time-invariant. We assume that $\eta(s_i) = w_l$ is given.

A plant set composed of N plants used for learning is represented by $\mathscr{H} = \{\eta_0, \eta_1, \ldots, \eta_{N-1}\}$. Set $\mathscr{H}$ corresponds to $\{\mathbf{P}_N + \Delta\mathbf{P}\}$. Let $\rho(\eta_n)$ denote the probability that the plant variation is η_n. We call this the existing probability of variation plant η_n. We assume that $\rho(\eta)$ is given at time $t = 0$.

For set $\mathscr{H}$, the expected cost of a policy μ starting from an initial state $s^0 = s_i$ at $t = 0$ is

$$\overline{J}^{\mu}(s_i) = \underset{\eta, s^1, s^2, \ldots}{E}\left[\sum_{t=t_0}^{\infty} g\left(s^t, \mu\left(s^t\right), s^{t+1}; \eta\left(s^t\right)\right) \mid s^0 = s_i,\ \eta \in \mathscr{H}\right], \tag{8}$$

which is the J-factor of this problem. This J-factor formula using the plant existing probability is

$$\overline{J}^{\mu}(s_i) = \sum_{n=0}^{N-1} \rho(\eta_n) J^{\mu,\eta_n}(s_i), \tag{9}$$

where $J^{\mu,\eta_n}(s_i)$ denotes the expected cost using the policy μ on a plant η_n starting from an initial state s_i. It satisfies

$$J^{\mu,\eta_n}(s_i) = \sum_{k=1}^{K} \pi(s_i, u_k) \sum_{j=0}^{N_s} p_{ij}(u_k; \eta_n(s_i)) \times \left\{g\left(s_i, u_k, s_j; \eta_n(s_i)\right) + J^{\mu,\eta_n}\left(s_j\right)\right\}. \tag{10}$$

We define the optimal policy as

$$\mu^*(s_i) \equiv \arg\min_{\mu} \sum_{i=1}^{N_s} \overline{J}^{\mu}(s_i), \quad \forall s_i, \tag{11}$$

which minimizes the J-factor of every state. The J-factor of the optimal policy μ^* is defined as the optimal J-factor, represented by $\overline{J}^*(s_i)$. The objective is to obtain the optimal policy. We assume that there is at least one policy μ satisfying $\overline{J}^{\mu}(s_i) < \infty$, $\forall s_i$, in this problem. Henceforth, we will call this problem the original problem.

The variation plant in the original problem correlates with differences between any two states. Due to this correlation, the optimal policy does not satisfy Bellman's principle of optimality [14]. Therefore, the optimal policy and the optimal J-factor in this problem do not satisfy (6) and (7). In general, the optimal policy is not stationary. If policies are limited to stationary, the optimal policy is stochastic.

Therefore, another problem definition or another solution method is needed.

6.3. Solutions for a Relaxed Problem of Reinforcement Learning with Variation. We relax the original problem to recover the principle of optimality. Then, we can find the optimal J-factor efficiently by applying DP algorithms to the relaxed problem. We treat a reinforcement learning problem based on a two-player zero-sum game.

We assume that differences, $w_0, w_1, \ldots, w_L$, exist independently in each state s_i. Then the original problem is relaxed to a reinforcement learning problem [15–17] based on a two-player zero-sum game [18] whose objective is to obtain the optimal policy for the worst variation maximizing the expected cost. Since the correlations of differences in a variation plant are ignored, the principle of optimality is recovered.

The $\mathscr{H}_{2pzs}$ is defined as the set of variation plants consisting of all possible combinations of any differences. Since each state has L types of differences, the number of variation plants in $\mathscr{H}_{2pzs}$ is L^{N_s}. We define the optimal policy μ^*_{2pzs} as the policy minimizing the expected cost against the worst variation η^*_{2pzs} maximizing the expected cost

$$\left(\mu^*_{2pzs}, \eta^*_{2pzs}\right) \equiv \arg\min_{\mu} \max_{\eta \in \mathscr{H}_{2pzs}} \sum_{i=1}^{N_s} J^{\mu,\eta}(s_i), \tag{12}$$

and the optimal J-factor $J^*_{2pzs}(s_i)$ is defined as the J-factor of the optimal policy and the worst variation.

Since the principle of optimality is recovered, the optimal J-factor satisfies the following Bellman equation:

$$J^*_{2pzs}(s_i) = \min_{u_k} \max_{w_l} \sum_{j=0}^{N_s} p_{ij}(u_k; w_l) \left\{ g(s_i, u_k, s_j; w_l) + J^*_{2pzs}(s_j) \right\}. \tag{13}$$

Therefore, the optimal J-factor can be obtained by a DP algorithm. Using the optimal J-factor, the optimal policy and the worst variation are obtained by

$$\left(\mu^*_{2pzs}(s_i), \eta^*_{2pzs}(s_i)\right) = \arg\min_{u_k} \max_{w_l} \sum_{j=0}^{N_s} p_{ij}(u_k; w_l) \times \left\{ g(s_i, u_k, s_j; w_l) + J^*_{2pzs}(s_j) \right\}. \tag{14}$$

The optimal policy μ^*_{2pzs} is applicable to all L^{N_s} variation plants in $\mathscr{H}_{2pzs}$. The optimal policy μ^*_{2pzs} is proper for all plants in $\mathscr{H}$ of Section 6.2 because $\mathscr{H} \subseteq \mathscr{H}_{2pzs}$ holds. However, the actual number of plants to which the policy should be applied is only N and $N \ll L^{N_s}$. Hence, the optimal policy of the reinforcement learning problem based on the two-player zero-sum game is often conservative and yields poor performance because the problem does not consider the existence of variation plants. We cannot solve this problem if there is no policy satisfying $J^{\mu,\eta}(s_i) < \infty, \forall s_i, \forall \eta \in \mathscr{H}_{2pzs}$, even though the policy μ exists and satisfies $\sum_i J^{\mu,\eta_n}(s_i) < \infty, \forall \eta_n \in \mathscr{H}$. Hence, a solution method to solve the original problem is desired.

6.4. Learning of Peg-in-Hole Task with Variation

6.4.1. Problem Definition of Peg-in-Hole Task with Variations. This section uses the same problem definition for peg-in-hole as Section 5.3.1. The following is added to take variations into account.

The hole position and orientation are measured by the stereo vision system. These measurements involve errors caused by quantized images, lighting conditions at a worksite, and so forth. Time-averaged errors are almost constant while the space robot performs the task, unlike white noise whose time-averaged error is zero. Error evaluations are modeled as described below. Hole position errors are modeled as normal probability distributions, where the mean is $m = 0$ [mm] and standard deviation is $\sigma = 0.75$ [mm]. Hole orientation errors are modeled as normal probability distributions, where the mean is $m = 0$ [rad] and standard deviation is $\sigma = 0.5\pi/180$ [rad]. If the error's statistical values gradually vary, we have to estimate them online. The relative position and orientation between Σ_0 and Σ_{hl} are fixed during the task. The plant variations are modeled as hole position and orientation measurement errors.

Consider these errors as variations $\Delta\mathbf{P}$ added to nominal model $\mathbf{P}_N$ (Figure 6(c)). We constructed 9 plants $\eta_0 \sim \eta_8$ for learning, as listed in Table 1, where each plant has a combination of errors among $[-1.0, 0.0, 1.0]$ [mm] in $\mathbf{j}_0$-axis direction and $[-(0.5/180)\pi, 0.0, (0.5/180)\pi]$ [rad] in $\mathbf{k}_0$-axis rotation. Plant η_0 with no error both in $\mathbf{j}_0$-axis direction and in $\mathbf{k}_0$-axis rotation is the nominal plant. The plant existing probabilities followed the above-mentioned normal probability distributions.

Table 1: Hole position of variation plant η_n from the nominal plant.

Plants	Variations	
	Position in $\mathbf{j}_0$ (mm)	Rotation about $\mathbf{k}_0$ (rad)
η_0	0.0	0.0
η_1	0.0	$-(0.5/180)\pi$
η_2	−1.0	0.0
η_3	−1.0	$-(0.5/180)\pi$
η_4	−1.0	$(0.5/180)\pi$
η_5	0.0	$(0.5/180)\pi$
η_6	1.0	0.0
η_7	1.0	$(0.5/180)\pi$
η_8	1.0	$-(0.5/180)\pi$

In the original problem, plant η_n determines the state transition probability as $p_{ij}(u_k; \eta_n)$ for all state transitions (s_i, u_k, s_j) simultaneously. The state transition probability is represented by $p_{ij}(u_k; w_n(s_i))$ where $\mathscr{W}(s_i) = \{w_0(s_i), \ldots, w_{N-1}(s_i)\}$ and $w_n(s_i) = \eta_n(s_i)$. On the other hand, the two-player zero-sum game allows difference w_l at state s_i to be chosen arbitrarily from $\mathscr{W}(s_i)$. The one-step cost is $g(s_i, u_k, s_j; w_l) = 1$.

In the later simulations and experiments to evaluate the learned results, the hole position and orientation are derived from the above normal probability distribution. In the simulations, a variation in the hole position and attitude is chosen for each episode, but the variation is invariant during the episode.

6.4.2. Learning Method. Under the conditions of the above problem definition, a policy is obtained by the solution in Section 6.3. It is the optimal policy μ^*_{2pzs} of the two-player zero-sum game, that is, $\mathbf{K}''_Q$ in Figure 6(c). The optimal policy μ^*_{2pzs} is applied to the dynamic simulator with a continuous state-action space or the experimental hardware setup, which is a real plant $\mathbf{P}$ in Figure 6(c). No online learning is needed.

There is no proper policy for all plants in $\mathscr{H}_{2pzs}$ if the variations in Table 1 are too large. In this case, there is no policy satisfying the reinforcement learning problem based on the two-player zero-sum game. A typical approach for this situation is to make the variations smaller, to reconstruct $\mathscr{H}_{2pzs}$, and to solve the two-player zero-sum game again. This approach is repeated if we cannot obtain any solutions. This approach reduces the robustness of solutions.

6.4.3. Control Results. In results for numerical simulation (Figure 10), where the peg arrives at the goal using controller $\mathbf{K}''_Q$ in the environment without hole position/orientation error. Peg positioning is firstly inclined, and the peg moves

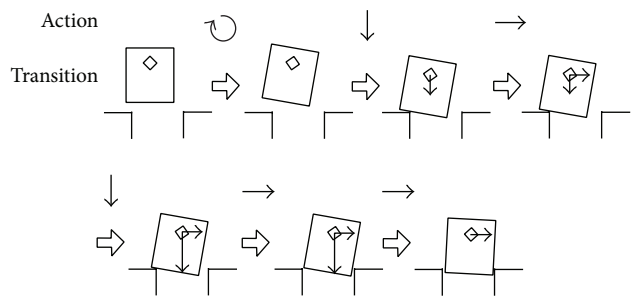


Figure 10: Trajectory of controller $\mathbf{K}_Q''$ in a simulation using nominal plant η_0 without any hole position error.

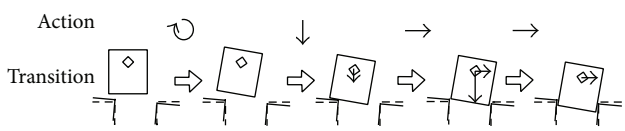


Figure 11: Trajectory of controller $\mathbf{K}_Q''$ in a simulation using plant η_1.

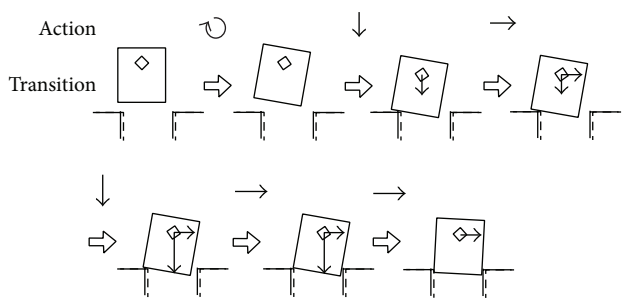


Figure 12: Trajectory of controller $\mathbf{K}_Q''$ in a simulation using plant η_2.

in the positive direction, toward $\mathbf{i}_0$. Then, the peg's corner is inserted in the hole. The peg makes contact with a corner of the hole. The peg transits in a positive direction, toward $\mathbf{j}_0$, while maintaining contact. Another corner of the peg is put inside the hole when the action in the direction of $\mathbf{i}_0$ and $\mathbf{j}_0$ is repeated. Peg positioning is changed to $y_\theta = 0$ [rad], and the peg slips into the hole. The task is completed. The learned result is similar to that of human skill for the peg-in-hole task in Figure 9.

The peg has arrived at the goal using controller $\mathbf{K}_Q''$ for variation plants η_1–η_8. The numerical results for η_1–η_4 are shown in Figures 11, 12, 13, and 14. Each transition is similar to the case of η_0, and the peg is inserted into the hole.

The task is achieved with controller $\mathbf{K}_Q''$ in the same environment with error, where $\mathbf{K}_Q$ previously did not work at all. This means that the action generated by controller $\mathbf{K}_Q''$ is robust against variations as well as human skill. We judge the robot, that is, controller $\mathbf{K}_Q''$, to have obtained a skill, the ability to complete a task when the vision measurement accuracy is low.

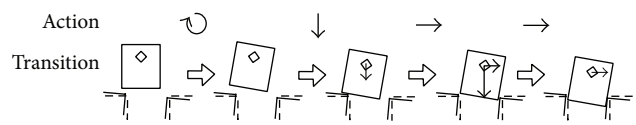


Figure 13: Trajectory of controller $\mathbf{K}_Q''$ in a simulation using plant η_3.

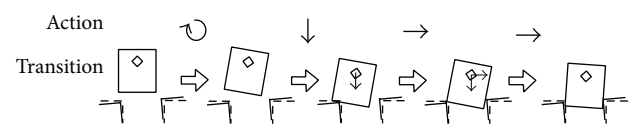


Figure 14: Trajectory of controller $\mathbf{K}_Q''$ in a simulation using plant η_4.

7. Evaluation of Obtained Skill

7.1. Results of Hardware Experiments. Example results in the hardware experiment using controllers $\mathbf{K}_Q$ and $\mathbf{K}_Q''$ are shown in Figure 15. The following variations are used: +0.3 [mm] in $\mathbf{i}_0$, +1.2 [mm] in $\mathbf{j}_0$, and $+0.5\pi/180$ [rad] rotation about the $\mathbf{k}_0$-axis. Controller $\mathbf{K}_Q$ cannot complete the task due to environmental variations, but controller $\mathbf{K}_Q''$ can.

7.2. Evaluation of Robustness and Control Performance. Robustness and control performance of controllers $\mathbf{K}_Q$ and $\mathbf{K}_Q''$ are evaluated by simulations and hardware experiments, the peg-in-hole task.

Variation plants, that is, error in hole position and orientation, are derived from the normal probability distribution in Section 6.4. The robustness and the control performance are evaluated, respectively, by the task achievement ratio and the average step number to the goal. The achievement ratio equals the number of successes divided by the number of trials. Table 2 shows the achievement ratios and the average step number of $\mathbf{K}_Q$ and $\mathbf{K}_Q''$ as evaluated by simulations and hardware experiments. The simulations and the experiments are executed 10,000 times and 50 times, respectively. The achievement ratios of $\mathbf{K}_Q$ are 59% and 64% in simulation and hardware experiments. Those of $\mathbf{K}_Q''$ dramatically increase to 99% in numerical simulation and 96% in hardware experiments. These results show that the robot autonomously

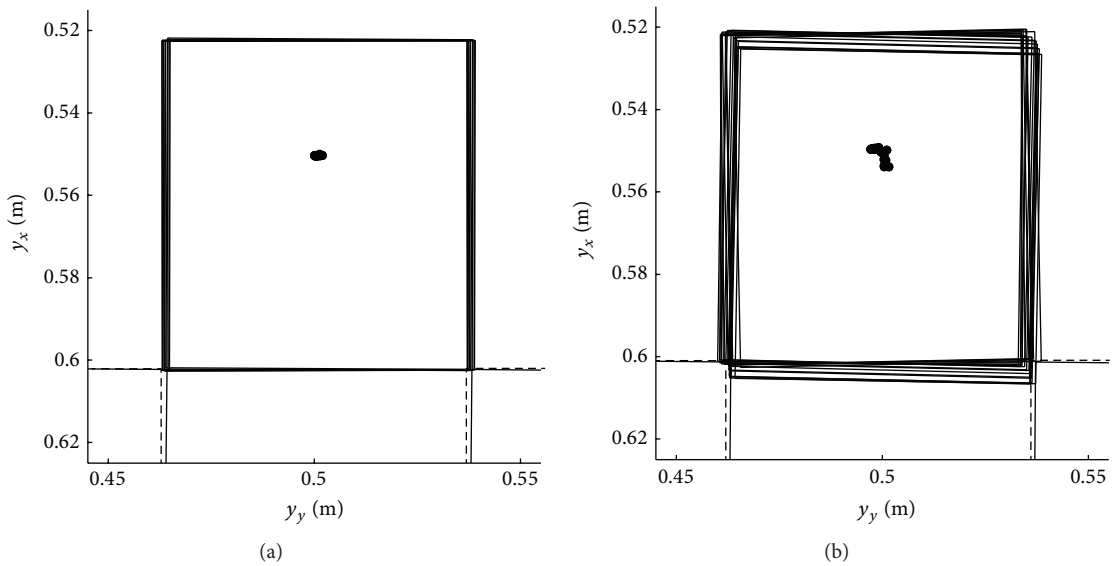


Figure 15: Experimental trajectories using two controllers in an environment with error (a) controller $\mathbf{K}_Q$ and (b) controller $\mathbf{K}''_Q$.

Table 2: Achievement ratios and averaged step numbers of peg-in-hole task with controllers $\mathbf{K}_Q$ and $\mathbf{K}''_Q$.

Controller	Simulation		Experiment	
	Ratio	Step number	Ratio	Step number
$\mathbf{K}_Q$	58.7%	19.3	64%	21
$\mathbf{K}''_Q$	98.7%	9.83	96%	17

generates robust skill using the proposed learning method. The difference in step numbers between hardware experiments and simulations, an increase in hardware steps, may be due to variations, for example, irregular friction in the environment, joint flexibility, and so forth. Such variables are not considered in the numerical simulation.

Robust skills are thus autonomously generated by learning in this situation, where variations make task achievement difficult.

8. Conclusions

We have applied reinforcement learning to obtain successful completion of a given task when a robot normally cannot complete the task using controller designed in advance. Peg-in-hole achievement ratios are usually low when we use conventional learning without consideration of plant variations. In the proposed method, using variation consideration, the robot autonomously obtains robust skills which enabled the robot to achieve the task. Simulation and hardware experiments have confirmed the effectiveness of our proposal. Our proposal also ensures robust control by conducting learning stages for a set of plant variations.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, Mass, USA, 1998.

[2] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific, 1996.

[3] S. Fujii, K. Senda, and S. Mano, "Acceleration of reinforcement learning by estimating state transition probability model," *Transactions of the Society of Instrument and Control Engineers*, vol. 42, no. 1, pp. 47–53, 2006 (Japanese).

[4] K. Senda, "An approach to autonomous space robots," *Systems, Control and Information*, vol. 45, no. 10, pp. 593–599, 2001 (Japanese).

[5] K. Senda, Y. Murotsu, A. Mitsuya et al., "Hardware experiments of a truss assembly by an autonomous space learning robot," *Journal of Spacecraft and Rockets*, vol. 39, no. 2, pp. 267–273, 2002.

[6] S. B. Skaar and C. F. Ruoff, Eds., *Teleoperation and Robotics in Space*, AIAA, Washington, DC, USA, 1995.

[7] M. Asada, "Issues in applying robot learning and evolutionary methods to real environments," *Journal of Society of Instrument & Control Engineers*, vol. 38, no. 10, pp. 650–653, 1999 (Japanese).

[8] D. E. Whitney, "Quasi-static assembly of compliantly supported rigid parts," *Journal of Dynamic Systems, Measurement and Control*, vol. 104, no. 1, pp. 65–77, 1982.

[9] D. Sato and M. Uchiyama, "Peg-in-hole task by a robot," *Journal of the Japan Society of Mechanical Engineers*, vol. 110, no. 1066, pp. 678–679, 2007 (Japanese).

[10] N. Yamanobe, Y. Maeda, T. Arai et al., "Design of force control parameters considering cycle time," *Journal of the Robotics Society of Japan*, vol. 24, no. 4, pp. 554–562, 2006 (Japanese).

[11] T. Fukuda, W. Srituravanich, T. Ueyama, and Y. Hasegawa, "A study on skill acquisition based on environment information (task path planning for assembly task considering uncertainty)," *Transactions of the Japan Society of Mechanical Engineers C*, vol. 66, no. 645, pp. 1597–1604, 2000 (Japanese).

[12] F. Miyazaki and S. Arimoto, "Sensory feedback for robot manipulators," *Journal of Robotic Systems*, vol. 2, no. 1, pp. 53–71, 1985.

[13] C. I. Connolly, J. B. Burns, and R. Weiss, "Path planning using Laplace's equation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2102–2106, May 1990.

[14] K. Senda and Y. Tani, "Optimality principle broken by considering structured plant variation and relevant robust reinforcement learning," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC '11)*, pp. 477–483, October 2011.

[15] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Proceedings of International Conference on Machine Learning*, pp. 157–163, 1994.

[16] J. Morimoto and K. Doya, "Robust reinforcement learning," *Neural Computation*, vol. 17, no. 2, pp. 335–359, 2005.

[17] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, "Model-free Q-learning designs for linear discrete-time zero-sum games with application to H-infinity control," *Automatica*, vol. 43, no. 3, pp. 473–481, 2007.

[18] T. Başar and P. Bernhard, *H_∞-Optimal Control and Related Minimax Design Problems*, Birkhäuser, Boston, Mass, USA, 1995.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2014, Article ID 237983, 10 pages
http://dx.doi.org/10.1155/2014/237983

Research Article

Second-Order Nonlinear Analysis of Steel Tapered Beams Subjected to Span Loading

Ali Hadidi, Bahman Farahmand Azar, and Hossein Zonouzi Marand

Department of Civil Engineering, University of Tabriz, Tabriz 51666-16471, Iran

Correspondence should be addressed to Ali Hadidi; alihadiditabriz@gmail.com

Received 6 August 2013; Accepted 4 September 2013; Published 13 March 2014

Academic Editor: Siamak Talatahari

A second-order elastic analysis of tapered steel members with I-shaped sections subjected to span distributed and concentrated loadings is developed. Fixed end forces and moments as well as exact stiffness matrix of tapered Timoshenko-Euler beam are obtained with exact geometrical properties of sections. The simultaneous action of bending moment, shear, and axial force including $P-\delta$ effects is also considered in the analysis. A computer code has been developed in MATLAB software using a power series method to solve governing second-order differential equation of equilibrium with variable coefficients for beams with distributed span loading. A generalized matrix condensation technique is then utilized for analysis of beams with concentrated span loadings. The accuracy and efficiency of the results of the proposed method are verified through comparing them to those obtained from other approaches such as finite element methods, which indicates the robustness and time saving of this method even for large scale frames with tapered members.

1. Introduction

Tapered members are used in many structures such as sloped frames, bridges, and multistory buildings as well as mechanical components. Because of the ease of construction, the use of I-shaped members with a linearly variable depth of web is more practical; moreover, weight optimization, stability increase (Timoshenko and Gere [1] and Lopez [2]), flexibility in fabrication and design, and sometimes satisfying architectural considerations are some advantages which rolled members cannot provide. Therefore, an exact analysis of structures containing this kind of members is very important. On the other hand, according to the AISC-2010 [3], the required strengths of components of structures should be determined by using a second-order analysis method. This kind of analysis can consider flexural, shear, and axial member deformations; both $P-\Delta$ and $P-\delta$ effects (Chen and Lui [4]); and all gravity and other applied loads that may influence the stability of the structure. Consequently, a direct analysis method for design of structures containing tapered members which are subject to nodal and span loading is of high significance.

In addition to classic methods, in recent decades, different approaches have been developed for second-order and buckling analysis of tapered members. Timoshenko and Young [5] suggested the division of the tapered members to subelements for analyzing the structure. Despite the simplicity of this method, they do not have the required precision for an exact analysis (as illustrated in example 1).

Numerical methods such as finite element method (e.g., Bathe [6]) and direct integral method (e.g., Karabalis and Beskos [7]) have been used for producing stiffness matrix of member. These methods achieve relatively good precision by choosing fine mesh in modeling; however, Numerical methods such as finite element method is highly time consuming due to its great number of finite elements, especially in the case of large structures. Therefore, these methods are not conducive to daily use because of high computational costs.

Al-Gahtani [8] obtained axial, torsional, and flexural stiffness matrix separately as well as fixed end forces and moments through solving governing equilibrium equations of member based on the boundary integral method. This study considers the span loadings on tapered beams; however,

it ignores the second-order effects associated with $P-\delta$ and simultaneous action of shear and bending deformation.

Li et al. [9–11] have solved tapered Timoshenko-Euler beam element using Chebyshev polynomial approach considering simultaneous effects of axial force, shear deformation, and $P-\delta$ effects with nodal loads. The exact stiffness matrix is derived based on beam-column theories taking into account second-order effects in governing equilibrium differential equation. This approach contains less computational effort and is also more time saving and practical in advanced analysis of structures; nevertheless, span loadings have not been considered in this method despite their importance in practical analysis.

This paper, alternatively, takes into account not only nodal loads but also distributed and concentrated span loads (Kim et al. [12]). A practical method for second-order analysis of steel tapered members is presented considering the simultaneous effect of bending moments, axial forces, and shear deformations (including $P-\delta$). To fulfill this purpose, a power series method is used to solve the equilibrium differential equation to obtain exact elastic stiffness matrix and fixed end forces and moments. In the offered examples, the effect of axial pressure forces and shear deformations is clearly observable in reduction of element stiffness. Although only elements with I-section and linear variation of depth are considered in this study, the proposed method can be developed for other states. The algorithm presented in this paper can be used for direct analysis method considering requirements based on AISC code [3].

2. Derivation of Element Formulation

The general form of members considered in this paper is shown in Figure 1 which reveals linearly symmetric tapered web and I-shaped section with constant width and thickness of flanges. It is assumed that the element is braced laterally while local buckling of the web and flanges is not taken into account. All elements are initially straight and the cross-sections of the beam remain plane after deformation. Stiffness matrix and fixed end forces and moments have been obtained for tapered members in this section. The element formulation is based on the exact moment of inertia of section with no approximation.

2.1. Geometrical Properties of Section. The exact formulation of area and inertia moment of the cross-section at location z as shown in Figure 1 is obtained as follows:

$$\begin{aligned} A_{(z)} &= 2b_f \cdot t_f + \left(D_{(z)} - 2t_f\right) \cdot t_w, \\ I_{(z)} &= \frac{1}{12} t_w \cdot D_{(z)}^3 + 2 \times \frac{1}{12} t_f^3 \cdot \left(b_f - t_w\right) \\ &\quad + 2t_f \cdot \left(b_f - t_w\right) \cdot \left(\frac{D_{(z)} - t_f}{2}\right)^2, \\ A_{w(z)} &= D_{(z)} \cdot t_w, \end{aligned} \tag{1}$$

where $D_{(z)} = D_1 + s \cdot z$, $s = (D_2 - D_1)/L$, and $A_{w(z)}$ is web area at location z and equal to overall depth time web thickness

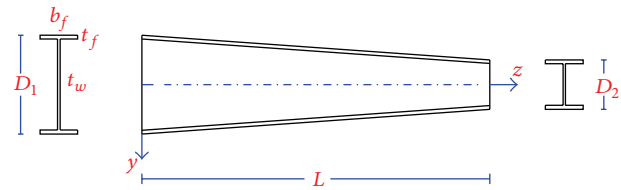


Figure 1: I-shaped section members with tapered web.

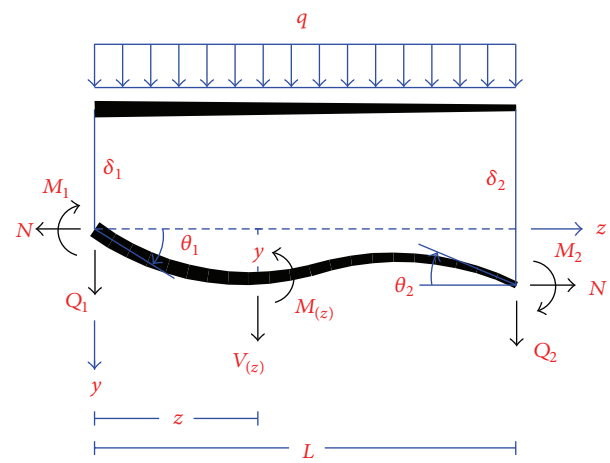


Figure 2: Tapered beam of the paper, the used conventions and uniform span loadings.

(AISC-2010 [3]). In most of the existing researches, approximate formulation has been used for calculation of inertia moment. This approximation creates considerable error in structural analysis results including displacements and forces. Some approximate formulas are shown in Appendix A.

2.2. Members Subject to Uniform Distributed Span Loading. Tapered beam-column element of this research including uniform distributed span loadings and applied conventions is shown in Figure 2.

In deriving stiffness matrix and fixed end forces and moments of tapered members, the simultaneous effect of bending moment and shear and axial force is considered. In so doing, the equilibrium differential equation of the tapered Timoshenko-Euler beam element is established, in a similar way to G. Q. Li and J. J. Li [9], in order to compare the obtained relations. Equation (2) is the governing equation for the equilibrium of tapered Timoshenko-Euler beams without span loadings (Li et al. [9–11]):

$$\begin{aligned} &\alpha_{(z)} \cdot y'' - \beta_{(z)} \cdot N \cdot y' - N \cdot y \\ &\quad = \beta_{(z)} \cdot Q_1 - \left(M_1 - Q_1 \cdot z\right) \quad 0 \le z \le L, \end{aligned} \tag{2}$$

where $\alpha_{(z)} = E \cdot I_{(z)} \cdot \gamma_{(z)}$, $\beta_{(z)} = E \cdot I_{(z)} \cdot (A'_{w(z)}/G \cdot A^2_{w(z)})$, and $\gamma_{(z)} = 1 + N/(G \cdot A_{w(z)})$.

E and G, respectively, are elastic and shear modulus. In this section, the governing equilibrium equation (2) is reestablished taking into account uniform span loadings.

Element deflection consists of two sections: one is induced by the bending deformation and the other by shear deformation; moreover, the axial deformation due to axial forces is taken into account separately:

$$y = y_M + y_Q. \tag{3}$$

Second derivation of (3) can be expressed as follows:

$$y'' = y''_M + y''_Q. \tag{4}$$

Substituting y''_M and y''_Q terms by relations obtained from (5) and rearranging terms for y'', y', and y, the governing equation for the equilibrium of tapered Timoshenko-Euler beam is obtained as (6):

$$\begin{gathered} M_{(z)} = M_1 - Q_1 \cdot z - M_{s(z)} - N \cdot y, \qquad Q_{(z)} = \frac{dM_{(z)}}{dz}, \\ y''_M = \frac{-M_{(z)}}{E \cdot I_{(z)}}, \qquad y'_Q = \frac{\mu \cdot Q_{(z)}}{G \cdot A_{(z)}}, \qquad \mu = \frac{A_{(z)}}{A_{w(z)}} \\ y'' = -\frac{M_1 - M_{s(z)} - Q_1 \cdot z - N \cdot y}{E \cdot I_{(z)}} + \frac{1}{G \cdot A_{w(z)}} \\ \times \left[-M''_{s(z)} - N \cdot y'' + \frac{A'_{w(z)}}{A_{w(z)}} \cdot \left(Q_1 + M'_{s(z)} + N \cdot y' \right) \right], \end{gathered} \tag{5}$$

$$\begin{aligned} &\alpha_{(z)} \cdot y'' - \beta_{(z)} \cdot N \cdot y' - N \cdot y \\ &\quad = \beta_{(z)} \cdot \left[\left(Q_1 + M'_{s(z)} \right) - M''_{s(z)} \cdot \frac{A_{w(z)}}{A'_{w(z)}} \right] \\ &\qquad - \left(M_1 - Q_1 \cdot z - M_{s(z)} \right) \quad 0 \le z \le L, \end{aligned} \tag{6}$$

where $M_{s(z)}$ is the moment created by span loadings at distance z from the left end of the element and $\alpha_{(z)} = E \cdot I_{(z)} \cdot \gamma_{(z)}$, $\beta_{(z)} = E \cdot I_{(z)} \cdot (A'_{w(z)}/G \cdot A^2_{w(z)})$, and $\gamma_{(z)} = 1 + N/(G \cdot A_{w(z)})$.

For converting (6) to nondimensional form let $\xi = z/L$:

$$\begin{aligned} &\alpha_{(\xi)} \cdot y'' - \beta_{(\xi)} \cdot L \cdot N \cdot y' - L^2 \cdot N \cdot y \\ &\quad = \beta_{(\xi)} \cdot L^2 \left[\left(Q_1 + M'_{s(\xi)} \right) - M''_{s(\xi)} \cdot \frac{A_{w(\xi)}}{A'_{w(\xi)}} \right] \\ &\qquad - L^2 \cdot \left(M_1 - Q_1 \cdot L \cdot \xi - M_{s(\xi)} \right) \quad 0 \le \xi \le 1. \end{aligned} \tag{7}$$

Equation (7) can be used for all tapered members with any sections and all forms of span loadings. Using (8) for uniform span loadings (q as in Figure 2) and I-shaped section, (7) is converted to (9); namely,

$$\begin{gathered} M_{s(z)} = \frac{1}{2} q \cdot z^2, \qquad M'_{s(z)} = q \cdot z, \qquad M''_{s(z)} = q \\ A_{w(z)} = \left(D_1 + s \cdot z \right) \cdot t_w, \qquad A'_{w(z)} = s \cdot t_w, \end{gathered} \tag{8}$$

$$\begin{aligned} &\alpha_{(\xi)} \cdot y'' - \beta_{(\xi)} \cdot L \cdot N \cdot y' - L^2 \cdot N \cdot y \\ &\quad = \beta_{(\xi)} \cdot L^2 \cdot \left(Q_1 - q \cdot \frac{D_1}{s} \right) \\ &\qquad - L^2 \cdot \left(M_1 - Q_1 \cdot L \cdot \xi - \frac{1}{2} q \cdot L^2 \cdot \xi^2 \right). \end{aligned} \tag{9}$$

To solve differential equation (9), power series method is used for y, α, and β:

$$\begin{gathered} y_{(\xi)} = \sum_{n=0}^{M} y_n \cdot \xi^n, \\ \alpha_{(\xi)} = \sum_{n=0}^{M} \alpha_n \cdot \xi^n, \\ \beta_{(\xi)} = \sum_{n=0}^{M} \beta_n \cdot \xi^n. \end{gathered} \tag{10}$$

Differentiating equation (10) term by term with respect to ξ and substituting them into (9), we arrive at the result and after a further differentiation we have

$$\begin{aligned} &\sum_{n=0}^{M} \left[\sum_{i=0}^{n} \alpha_i \left(n + 2 - i \right) \left(n + 1 - i \right) y_{n+2-i} \right] \\ &\quad \cdot \xi^n - L \cdot N \cdot \sum_{n=0}^{M} \left[\sum_{i=0}^{n} \beta_i \left(n + 1 - i \right) y_{n+1-i} \right] \\ &\quad \cdot \xi^n - L^2 \cdot N \cdot \sum_{n=0}^{M} y_n \cdot \xi^n \\ &\quad = L^2 \cdot \left(Q_1 - q \cdot \frac{D_1}{s} \right) \cdot \sum_{n=0}^{M} \beta_n \cdot \xi^n \\ &\qquad - L^2 \cdot M_1 + L^3 \cdot Q_1 \cdot \xi + \frac{1}{2} q \cdot L^4 \cdot \xi^2. \end{aligned} \tag{11}$$

Separating out the terms corresponding to $n = 0$, $n = 1$, and $n \ge 2$, and then the equalization of relevant coefficients in two sides of (11), for $n = 0$, it becomes:

$$\begin{aligned} &2\alpha_0 \cdot y_2 - L \cdot N \cdot \beta_0 \cdot y_1 - L^2 \cdot N \cdot y_0 \\ &\quad = L^2 \cdot \beta_0 \left(Q_1 - q \cdot \frac{D1}{s} \right) - L^2 \cdot M_1, \end{aligned} \tag{12a}$$

for $n = 1$,

$$\begin{aligned} &6\alpha_0 \cdot y_3 + 2\alpha_1 \cdot y_2 - L \cdot N \left(2\beta_0 \cdot y_2 + \beta_1 \cdot y_1 \right) \\ &\quad - L^2 \cdot N \cdot y_1 = L^2 \cdot \beta_1 \left(Q_1 - q \cdot \frac{D_1}{s} \right) + L^3 \cdot Q_1, \end{aligned} \tag{12b}$$

and finally for $n \geq 2$,

$$\begin{aligned}
&\alpha_0 (n+2)(n+1) y_{n+2} + \sum_{i=1}^{n} \alpha_i (n+2-i)(n+1-i) y_{n+2-i} \\
&\quad - L \cdot N \cdot \sum_{i=0}^{n} \beta_i (n+1-i) y_{n+1-i} - L^2 \cdot N \cdot y_n \\
&\quad = L^2 \cdot \beta_n \left(Q_1 - q \cdot \frac{D_1}{s} \right) + 2q \cdot L^3 \cdot \beta_{n-1} + \frac{1}{2} q \cdot L^4 \left[\frac{2}{n} \right],
\end{aligned} \tag{13}$$

where $[2/n] = 1$, for $n = 2$, and $[2/n] = 0$, for $n > 2$.

Recurrence relation for $n \geq 2$ using (13) can be written as

$$\begin{aligned}
y_{n+2} = \Bigg(\Bigg[& L \cdot N \cdot \sum_{i=0}^{n} \beta_i (n+1-i) y_{n+1-i} \\
& + L^2 \cdot N \cdot y_n + L^2 \cdot \beta_n \left(Q_1 - q \cdot \frac{D_1}{s} \right) \\
& + 2q \cdot L^3 \cdot \beta_{n-1} + \frac{1}{2} q \cdot L^4 \left[\frac{2}{n} \right] \\
& - \sum_{i=1}^{n} \alpha_i (n+2-i)(n+1-i) y_{n+2-i} \Bigg] \Bigg) \\
& \times \left(\alpha_0 (n+2)(n+1) \right)^{-1}.
\end{aligned} \tag{14}$$

Thus, considering (14), it can be concluded that any y_n (for $n \geq 4$) can be expressed in linear combination of y_0, y_1, y_2, y_3, Q_1 and the constant value of c_5 as

$$y = c_0 \cdot y_0 + c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 + c_4 \cdot Q_1 + c_5. \tag{15}$$

Hereafter, these parameters must be obtained to determine y for different values of z. It should also be mentioned that M_1 is unknown and appears in y_n (for $n < 4$). From (5) and considering $y' = y'_M + y'_Q$ and $y'_M = \theta_\xi$,

$$y'(\xi) = \frac{L}{\gamma(\xi)} \left[\theta_\xi - \frac{Q_1 + q \cdot L \cdot \xi}{G \cdot A_w(\xi)} \right]. \tag{16}$$

Using boundary condition, (15), and (16).

For $\xi = 0$,

$$\begin{aligned}
y(0) &= y_0 = 0, \\
y'(0) &= y_1 = \frac{L}{\gamma(0)} \left[\theta_1 - \frac{Q_1}{G \cdot A_w(0)} \right].
\end{aligned} \tag{17}$$

For $\xi = 1$,

$$\begin{aligned}
y(1) &= c_1 y_1 + c_2 y_2 + c_3 y_3 + c_4 Q_1 + c_5 = \delta_2 - \delta_1 \\
y'(1) &= c_6 y_1 + c_7 y_2 + c_8 y_3 + c_9 Q_1 \\
&\quad + c_{10} = \frac{L}{\gamma(1)} \left[\theta_2 - \frac{Q_1}{G \cdot A_w(1)} \right].
\end{aligned} \tag{18}$$

Using first relation of (10), separating out the terms corresponding to $n < 4$, collecting all the remaining terms under a single summation sign, and considering (18),

$$\begin{aligned}
y(1) &= y_1 + y_2 + y_3 \\
&\quad + \sum_{i=4}^{m} y_i = c_1 y_1 + c_2 y_2 + c_3 y_3 + c_4 Q_1 + c_5, \\
y'(1) &= y_1 + 2y_2 + 3y_3 \\
&\quad + \sum_{i=4}^{m} i y_i = c_6 y_1 + c_7 y_2 + c_8 y_3 + c_9 Q_1 + c_{10}.
\end{aligned} \tag{19}$$

Applying the below conditions to (19), coefficients c_1 to c_{10} are derived:

$$\begin{aligned}
&\text{condition } (0): y_1 = y_2 = y_3 = Q_1 = 0: \\
&\quad \sum_{i=4}^{m} y_i^{(0)} = c_5, \qquad \sum_{i=4}^{m} i y_i^{(0)} = c_{10} \\
&\text{condition } (1): y_1 = 1,\ y_2 = y_3 = Q_1 = 0: \\
&\quad 1 + \sum_{i=4}^{m} y_i^{(1)} = c_1 + c_5, \qquad 1 + \sum_{i=4}^{m} i y_i^{(1)} = c_6 + c_{10} \\
&\text{condition } (2): y_2 = 1,\ y_1 = y_3 = Q_1 = 0: \\
&\quad 1 + \sum_{i=4}^{m} y_i^{(2)} = c_2 + c_5, \qquad 2 + \sum_{i=4}^{m} i y_i^{(2)} = c_7 + c_{10} \\
&\text{condition } (3): y_3 = 1,\ y_1 = y_2 = Q_1 = 0: \\
&\quad 1 + \sum_{i=4}^{m} y_i^{(3)} = c_3 + c_5, \qquad 3 + \sum_{i=4}^{m} i y_i^{(3)} = c_8 + c_{10} \\
&\text{condition } (4): Q_1 = 1,\ y_1 = y_2 = y_3 = 0: \\
&\quad \sum_{i=4}^{m} y_i^{(4)} = c_4 + c_5, \qquad \sum_{i=4}^{m} i y_i^{(4)} = c_9 + c_{10}.
\end{aligned} \tag{20}$$

Now (12a), (12b), (17), and (18) can be used for obtaining unknown factors y_0, y_1, y_2, y_3, Q_1, and M_1 in order to find function $y_{(z)}$ for all values of z. Q_2 and M_2 can be found from equilibrium of element as

$$\begin{aligned}
Q_2 &= -Q_1 - q \cdot L, \\
M_2 &= -M_1 + Q_1 \cdot L + N \cdot (\delta_2 - \delta_1) + \frac{1}{2} q \cdot L^2.
\end{aligned} \tag{21}$$

On the other hand axial stiffness matrix of element can be derived separately as

$$[K_a] = \begin{bmatrix} k_a & -k_a \\ -k_a & k_a \end{bmatrix}, \tag{22}$$

where $k_a = E / \int_0^L (1/A(z)) dz$.

Solving the obtained relations for δ_1, θ_1, δ_2, and θ_2 yields stiffness matrix and fixed end forces and moments in the following form:

$$\{P\} = [K]\{\Delta\} + \left\{P^F\right\}, \tag{23}$$

where

$$[K] = \begin{bmatrix} k_a & 0 & 0 & -k_a & 0 & 0 \\ 0 & k_{11} & k_{12} & 0 & k_{13} & k_{14} \\ 0 & k_{21} & k_{22} & 0 & k_{23} & k_{24} \\ -k_a & 0 & 0 & k_a & 0 & 0 \\ 0 & k_{31} & k_{32} & 0 & k_{33} & k_{34} \\ 0 & k_{41} & k_{42} & 0 & k_{43} & k_{44} \end{bmatrix},$$

$$\{\Delta\} = \begin{bmatrix} u_1 \\ \delta_1 \\ \theta_1 \\ u_2 \\ \delta_2 \\ \theta_2 \end{bmatrix}, \quad \left\{P^F\right\} = \begin{bmatrix} 0 \\ Q_1^F \\ M_1^F \\ 0 \\ Q_2^F \\ M_2^F \end{bmatrix}, \quad \{P\} = \begin{bmatrix} N \\ Q_1 \\ M_1 \\ N \\ Q_2 \\ M_2 \end{bmatrix}. \tag{24}$$

It should be mentioned that, by substituting (21) into (23), we have

$$\begin{aligned} [k_{31} \;\; k_{32} \;\; k_{33} \;\; k_{34}] &= [-k_{11} \;\; -k_{12} \;\; -k_{13} \;\; -k_{14}] \\ [k_{41} \;\; k_{42} \;\; k_{43} \;\; k_{44}] &= [\, -k_{21} + k_{11} \cdot L - N \;\; -k_{22} + k_{12} \cdot L \\ &\qquad -k_{23} \;\; +k_{13} \cdot L \;\; +N \;\; -k_{24} \;\; +k_{14} \cdot L \,] \\ Q_2^F &= -Q_1^F - q \cdot L, \qquad M_2^F = -M_1^F + Q_1^F \cdot L + \frac{1}{2} q \cdot L^2. \end{aligned} \tag{25}$$

In (24) $[K]$, $\{\Delta\}$, $\{P\}$, and $\{P^F\}$ are elastic stiffness matrix of element, displacement vector, force vector, and fixed end forces and moments vector, respectively; u_1 and u_2 are axial displacement of element nodes. Component of stiffness matrix $[K]$ and vector $\{P^F\}$ are represented in Appendix B. Stiffness matrix is symmetric and there is no need for calculating all components of it. This subject is illustrated in example 1.

2.3. Members Subject to Concentrated Span Loading. In this part, a static condensation method is used for obtaining the stiffness matrix and fixed end forces and moments of the element. McGuire et al. [13] contracted the stiffness matrix of an element with eliminating certain degrees of freedom. Similar approach is used to establish formulation of the element with concentrated span loadings. To this purpose, the element is divided into two parts at the concentrated load location, as in Figure 3, and the stiffness matrix of each part is calculated from (24). Then, displacement quantities at division section are eliminated so it becomes possible to calculate fixed end forces and moments independently of the intermediate displacement vector. Consequently, this approach requires less number of arithmetical operations than those with no condensing. It should be mentioned that the stiffness matrix obtained from the proposed approach is the same as that calculated through (24). But in the presence of concentrated span loadings, fixed end forces and moments and the stiffness matrix of element are obtained. The proposed formulation is established as follows.

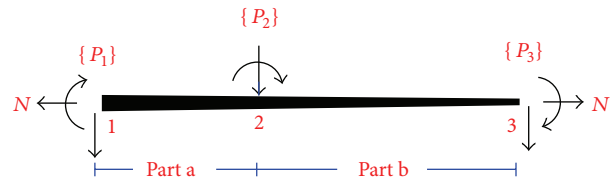


Figure 3: Divided tapered beam with concentrated span loadings.

The force-displacement relation of parts (a) and (b) is $[K_1]\{\Delta_1\} = \{P_1\}$ and $[K_2]\{\Delta_2\} = \{P_2\}$, respectively, where

$$\begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k'_{22} \end{bmatrix} \begin{Bmatrix} \Delta_1 \\ \Delta_2 \end{Bmatrix} = \begin{Bmatrix} P_1 \\ P'_2 \end{Bmatrix}, \tag{26}$$

$$\begin{bmatrix} k''_{22} & k_{23} \\ k_{32} & k_{33} \end{bmatrix} \begin{Bmatrix} \Delta_2 \\ \Delta_3 \end{Bmatrix} = \begin{Bmatrix} P''_2 \\ P_3 \end{Bmatrix}. \tag{27}$$

$[K_1]$ and $[K_2]$ are stiffness matrix of parts (a) and (b), obtained by (24).

Assembling (26) and (27) together, we have

$$\begin{bmatrix} k_{11} & k_{12} & 0 \\ k_{21} & k_{22} & k_{23} \\ 0 & k_{32} & k_{33} \end{bmatrix} \begin{Bmatrix} \Delta_1 \\ \Delta_2 \\ \Delta_3 \end{Bmatrix} = \begin{Bmatrix} P_1 \\ P_2 \\ P_3 \end{Bmatrix}, \tag{28}$$

in which,

$$\begin{aligned} P_2 &= P'_2 + P''_2, \\ k_{22} &= k'_{22} + k''_{22} \end{aligned} \tag{29}$$

is desired.

After solving recent relation for Δ_2 and substituting it in the first and third rows of (28) the following equation is obtained which is independent of Δ_2:

$$\begin{Bmatrix} P_1 \\ P_3 \end{Bmatrix} = \begin{bmatrix} k_{11} - k_{12}{k_{22}}^{-1}k_{21} & -k_{12}{k_{22}}^{-1}k_{23} \\ -k_{32}{k_{22}}^{-1}k_{21} & k_{33} - k_{32}{k_{22}}^{-1}k_{23} \end{bmatrix} \begin{Bmatrix} \Delta_1 \\ \Delta_3 \end{Bmatrix} + \begin{Bmatrix} k_{12}{k_{22}}^{-1}P_2 \\ k_{32}{k_{22}}^{-1}P_2 \end{Bmatrix}. \tag{30}$$

Table 1: Maximum deflection of fixed-hinged beam (mm).

Method	Case A	Case B	Case C	Case D	Location (mm)*
Proposed method	2.085 (1.8%)**	2.065	1.448	1.438	1845
Subdivided beam	2.210 (7.9%)	2.189	1.530	1.507	1800
FEM mesh 50 mm	2.047 (0.05%)	2.038	1.396	1.389	1844
FEM mesh 10 mm	2.048 (–)	2.039	1.406	1.400	1844

Case A: both axial and shear; B: no axial; C: no shear; D: no axial nor shear.
*Maximum deflection location from first of beam (mm).
**Errors: calculated compared to FEM with 10 mm mesh size.

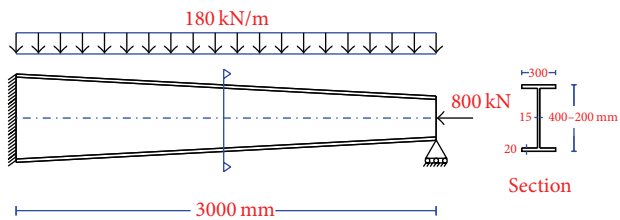


Figure 4: Tapered beam with uniformly distributed span loadings.

Obtained relation is in the form of $\{P\} = [K]\{\Delta\} + \{P^F\}$, where $\{P^F\}$ is fixed end forces and moments:

$$\left\{P^F\right\} = \begin{Bmatrix} k_{12}k_{22}{}^{-1}P_2 \\ k_{32}k_{22}{}^{-1}P_2 \end{Bmatrix}. \tag{31}$$

And stiffness matrix $[K]$ is

$$[K] = \begin{bmatrix} k_{11} - k_{12}k_{22}{}^{-1}k_{21} & -k_{12}k_{22}{}^{-1}k_{23} \\ -k_{32}k_{22}{}^{-1}k_{21} & k_{33} - k_{32}k_{22}{}^{-1}k_{23} \end{bmatrix}. \tag{32}$$

Example 2 indicated that matrix $[K]$ is the same stiffness matrix of whole member with no division. So it is enough to calculate k_{12}, k_{22}, and k_{32} to find fixed end forces and moments. It should be mentioned that axial stiffness matrix components can be calculated in a similar way to that in the previous section.

3. Numerical Examples and Verification

This section offers some examples to verify the accuracy of obtained relations; in this measure, the deflection and normal stress of beam-column elements in the proposed method are compared with the results of other studies and finite element method. The examples also reveal the symmetry of the stiffness matrix.

3.1. Beam Column Subject to Uniformly Distributed Span Loadings. The tapered fixed-hinged beam in Figure 4 is similar to G. Q. Li and J. J. Li example [9]; however, in this study it is subject to uniform distributed span loadings where maximum deflection and its location are calculated. Table 1 shows the comparison of the results of the proposed method obtained in MATLAB [14] software to those of FEM with 50 and 10 mm mesh size and also to the beam subdivided into five segments of equivalent nodal loads. It also indicates the effect of pressure axial force and shear deflection. To generate the results for verification, the ABAQUS [15] four-node shell element S4R is employed for modeling the beam as in Figure 5. The results demonstrate an acceptable accuracy of the proposed method. The investigations revealed that choosing number 14 for polynomial term is acceptable in calculating deflections. In all examples of this paper, elastic and shear modulus are 206 and 80 KN/mm^2, respectively.

Stiffness matrix $[K]$ and fixed end forces and moments $\{P^F\}$ are shown below; it can be seen that matrix $[K]$ is symmetric and also reduces computational efforts of calculation:

$$[K] = 10^6 \begin{bmatrix} 1.1092 & 0 & 0 & -1.1092 & 0 & 0 \\ 0 & 0.0183 & 34.9736 & 0 & -0.0183 & 20.7742 \\ 0 & 34.9679 & 80920.2118 & 0 & -34.9679 & 23983.4320 \\ -1.1092 & 0 & 0 & 1.1092 & 0 & 0 \\ 0 & -0.0183 & -34.9736 & 0 & 0.0183 & -20.7742 \\ 0 & 20.7799 & 24000.6566 & 0 & -20.7799 & 38339.1768 \end{bmatrix}$$

$$\{P\} = 10^6 \begin{bmatrix} -.8000 \\ -.3505 \\ -241.6648 \\ -.8000 \\ -0.1894 \\ 0 \end{bmatrix} \quad \{\Delta\} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -0.7212 \\ 0 \\ -2.5145 \times 10^{-3} \end{bmatrix} \quad \left\{P^F\right\} = 10^6 \begin{bmatrix} 0 \\ -0.2983 \\ -181.3579 \\ 0 \\ -0.2417 \\ 96.4047 \end{bmatrix}. \tag{33}$$

Table 2: Deflection at middle of fixed-hinged beam (mm).

Method	Case A	Case B	Case C	Case D
Proposed method	0.639 (0.2%)*	0.636	0.409	0.407
FEM mesh 300 mm	0.642 (0.6%)	0.640	0.367	0.365
FEM mesh 50 mm	0.637 (0.2%)	0.635	0.398	0.396
FEM mesh 10 mm	0.638	0.636	0.401	0.399
Li (2 elem.)	0.672 (5.3%)	0.669	0.409	0.408
(10 elem.)	0.673 (5.5%)	0.671	0.411	0.410

Case A: both axial and shear; B: no axial; C: no shear; D: no axial nor shear.
*Errors: calculated compared to FEM with 10 mm mesh.

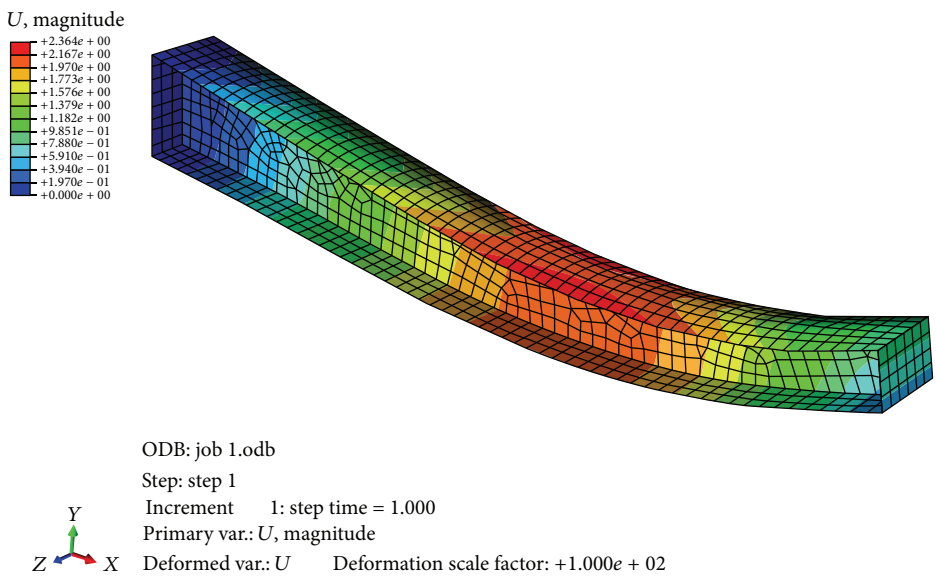


Figure 5: Finite element model of tapered beam.

3.2. Beam Column Subject to Concentrated Span Loading. Calculating the deflection at midspan of steel tapered fixed-hinged beam of G. Q. Li and J. J. Li [9] is considered in Figure 6. The results of the proposed method are compared to those of FEM with 50, 20, and 10 mm mesh size and those of Li et al. study (Table 2). G. Q. Li and J. J. Li divided the member into two parts and derived stiffness matrix of each part individually; they also utilized FEM by dividing the element into 10 segments in its length to calculate deflection. Because of the same axial loads in the two parts of the beam while dealing with elastic analysis, the stiffness matrix of the whole member in this example is the same as that of the previous one.

3.3. Gable Frame with Tapered Members and Span Loadings. This section takes into account a very applicable single bay gable frame shown in Figure 7; similar to previous examples the results were verified by ABAQUS with 50 mm mesh size. To this purpose, maximum deflection and normal stress at the tip and corner of the frame and also the effects of shear

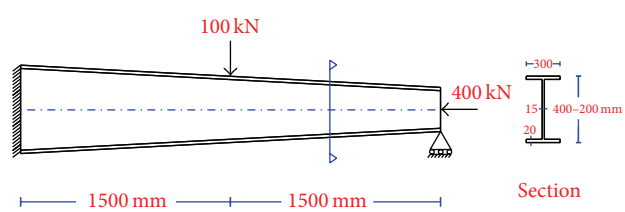


Figure 6: Tapered beam with concentrated span loadings.

deformations and $P - \delta$ on deflections are taken into consideration (Table 3). These results evidently expose the efficiency of the proposed method in practical problems. It should also be mentioned that, in the proposed method, the elements are modeled symmetrically over their longitudinal axes.

4. Conclusions

This paper investigated second-order nonlinear analysis of tapered Timoshenko-Euler beam-column element subjected

Table 3: Response of gable frame.

Response	Method	Case A		B	C
		Tip	Corner		
Deflection (mm)	Proposed method	11.12 (1.7%)*	2.57 (0.4%)	11.09	10.40
	FEM mesh 50 mm	10.93	2.58	10.91	10.24
Max. stress (N/mm^2)	Proposed method	53.52 (5.3%)*	58.06 (0.5%)	—	—
	FEM mesh 50 mm	50.83	57.75	—	—

Case A: both axial and shear; B: no axial; C: no shear.
*Errors: calculated compared to FEM with 50 mm mesh.

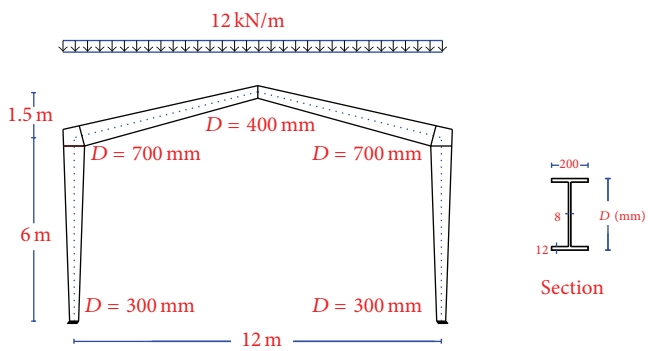


Figure 7: Gable frame with tapered members and span loadings.

to span loadings. Thus, a power series method was used to solve governing differential equations of the member equilibrium. According to examined examples, the appropriate number of terms for power series became 14 to achieve adequate precision and speed. The effect of shear displacements and pressure axial forces in the reduction of beam-column element stiffness is obviously recognizable in the solved examples. The high speed of the proposed method, its adequate precision and accuracy, and its simplicity in modeling make it a more effective and practical method compared to others such as finite element method. The required analysis time of the proposed approach for the first example is only 0.09 seconds, while it is 0.7 and 17 seconds for the finite element method with 50 and 10 mm meshes, respectively. The offered method can be applied in 2D large scale practical structures and also be generalized to spatial ones. Moreover, it can be developed in various mechanical component analyses such as rotating flexible taper beams and roof-crane taper bridges.

The proposed formulation can also be derived for each section shape, nonlinear variation of the section along member, and arbitrary distribution of span loadings. Therefore, member weight can also be considered in the analysis. Future work should include material nonlinearity and other conditions of steel design codes in order to achieve more advanced and direct analysis of frames including tapered members. Also, reliability analysis of tapered mechanical components subjected to harmonic excitations can be accomplished, like Rahbari et al. [16].

Appendices

A. Some Approximate Formulas for Moment of Inertia

(a) $I_z = I_f(z/a)^2$, where $a = (\sqrt{I_f/I_e}/(1-\sqrt{I_f/I_e}))\cdot L$ and I_f and I_e are moment of inertia at the ends of the member (Saffari et al. [17]);

(b) $I_z = I_0(1+cz/L)^{n+2}$, where I_0 is moment of inertia at the origin of element and c and n depend on degree of tapering and shape of the cross-section. (Al-Gahtani [8]);

(c) $I_z = I_1(1+(H_2/H_1-1)z/L)^n$, where $n = \log(I_2/I_1)/\log(H_2/H_1)$ (King at el. [18]);

(d) $I_z = (\phi_1+\phi_2 z)^2(\phi_3+\phi_4 z)$, $\phi_1 = D_1 - t_f$, $\phi_2 = (D_2 - D_1)/l$, $\phi_3 = t_w\phi_1/12 + b_f t_f/2$, and $\phi_4 = t_w\phi_2/12$ (Saka [19]);

(e) $I_z = I_f + (I_e - I_f)(z/L)^n$, $n = 1, 2, 3$, determined by the user (SAP 2000 [20]).

B. Component of Stiffness Matrix and Fixed End Forces and Moments

One has

$$k_a = \frac{E}{\int_0^L (1/A(z))\,dz}$$

$$k_{11} = \frac{-\psi_4}{\phi_1}, \qquad k_{12} = \frac{L\psi_5}{\psi_{10}},$$

$$k_{13} = -k_{11}, \qquad k_{14} = \frac{-L(6c_2\alpha_0 - c_3\psi_1)}{\psi_{11}}$$

$$k_{21} = \frac{\phi_2}{(\phi_1 L^2)}, \qquad k_{22} = \frac{-\phi_4}{(L\psi_{10})},$$

$$k_{23} = -k_{21}, \qquad k_{24} = \frac{\phi_3}{(L\psi_{11})}$$

$$[k_{31}\ \ k_{32}\ \ k_{33}\ \ k_{34}] = [-k_{11}\ \ -k_{12}\ \ -k_{13}\ \ -k_{14}]$$

$$
\begin{aligned}
&[k_{41} \quad k_{42} \quad k_{43} \quad k_{44}] \\
&= [\ -k_{21} + k_{11} \cdot L - N \quad -k_{22} + k_{12} \cdot L \\
&\qquad -k_{23} + k_{13} \cdot L + N \quad -k_{24} + k_{14} \cdot L\ ] \\
Q_1^F &= \left(-c_3 q \psi_1 L^2 - 6c_2 q \alpha_0 L^2 + (c_3 c_{10} - c_7 c_9)\psi_1 \psi_{13}\right. \\
&\qquad \left. + 6\alpha_0 \psi_{13}(c_6 c_9 - c_2 c_{10}) + q\psi_{13}\beta_1 D L^2 (c_2 c_7 - c_3 c_6)\right) \\
&\qquad \times (\phi \psi_{13})^{-1} \\
M_1^F &= \frac{\left(c_9 \phi_2 - \phi_3 c_{10} - q\phi_3 L^2/\psi_{13}\right)}{(\phi_1 L^2)} - \beta_0 q D \\
&\qquad + \left(\beta_1 D q 2\alpha_1 c_3 \left(c_8 + \frac{L}{\psi_{13}}\right)\right. \\
&\qquad - 2\alpha_0 c_4 c_7 + \psi_9 (c_3 c_6 - c_2 c_7) \\
&\qquad \left. + \frac{2\alpha_0 L (c_1 c_7 - c_3 c_5)}{\psi_{12}} - \frac{\psi_9 N (c_2 c_7 + c_3 c_6)}{\psi_{12}}\right) \\
&\qquad \times (\phi_1)^{-1} \\
Q_2^F &= -Q_1^F - q \cdot L, \qquad M_2^F = -M_1^F + Q_1^F \cdot L + \frac{1}{2} q \cdot L^2 \\
\psi_1 &= 2\alpha_1 - 2\psi_8, \qquad \psi_2 = 6\alpha_0 c_6, \qquad \psi_3 = N \cdot L (L + \beta_1), \\
\psi_4 &= \psi_2 - c_7 \psi_1 \\
\psi_5 &= (c_2 c_7 - c_3 c_6)\psi_3 + (c_1 c_7 - c_3 c_5)\psi_1 + 6\alpha_0 (c_2 c_5 - c_1 c_6), \\
\psi_6 &= c_8 + \frac{L}{\psi_{13}}, \qquad \psi_7 = L^2 \cdot (L + \beta_1) \\
\psi_8 &= \beta_0 \cdot L \cdot N, \qquad \psi_9 = \beta_0 \cdot L^2, \qquad \psi_{10} = \phi_1 \gamma(0), \\
\psi_{11} &= \phi_1 \gamma(1), \qquad \psi_{12} = A_w(0) \cdot G \cdot \gamma(0), \\
\psi_{13} &= A_w(1) \cdot G \cdot \gamma(1) \\
\phi_1 &= c_3 \psi_1 \psi_6 - 6\alpha_0 c_2 \psi_6 - c_4 c_7 \psi_1 + 6\alpha_0 c_4 c_6 \\
&\qquad - c_2 c_7 \psi_7 + c_3 c_6 \psi_7 + \frac{L (c_2 c_7 - c_3 c_6)\psi_3}{\psi_{12}} \\
&\qquad + \frac{L (c_1 c_7 - c_3 c_5)\psi_1}{\psi_{12}} - \frac{6\alpha_0 L (c_1 c_6 - c_2 c_5)}{\psi_{12}} \\
\phi_2 &= c_7 \psi_1 \psi_9 - 2\alpha_0 c_7 \psi_7 - 12{\alpha_0}^2 \psi_6 - 6\alpha_0 c_6 \psi_9 \\
&\qquad + \frac{12L{\alpha_0}^2 c_5}{\psi_{12}} + \frac{2L\alpha_0 c_7 \psi_{13}}{\psi_{12}} - \frac{N c_7 \psi_1 \psi_9}{\psi_{12}} + \frac{6N\alpha_0 c_6 \psi_9}{\psi_{12}} \\
\phi_3 &= c_3 \psi_9 \psi_1 - 2\alpha_0 (c_3 \psi_7 + 3c_2 \psi_9) \\
&\qquad - 12\alpha_0^2 \left(c_4 - \frac{c_1 L}{\psi_{12}}\right) + \frac{2\alpha_0 c_3 L \psi_{13}}{\psi_{12}} \\
&\qquad - \frac{c_3 N \psi_9 \psi_1}{\psi_{12}} + \frac{6\alpha_0 c_2 N \psi_9}{\psi_{12}} \\
\phi_4 &= 12\alpha_0^2 (c_1 \psi_6 - c_4 c_5) + 2\alpha_0 (c_3 \psi_6 \psi_3 - c_4 c_7 \psi_3 \\
&\qquad + c_1 c_7 \psi_7 - c_3 c_5 \psi_7) \\
&\qquad - (c_2 c_7 - c_3 c_6)\psi_3 \psi_9 - (c_1 c_7 - c_3 c_5)\psi_1 \psi_9 \\
&\qquad + 6\alpha_0 \psi_9 (c_1 c_6 - c_2 c_5) - c_3 \psi_1 \psi_6 \psi_8 \\
&\qquad + 6\alpha_0 \psi_8 (c_2 \psi_6 - c_4 c_6) + c_4 c_7 \psi_1 \psi_8 \\
&\qquad + (c_2 c_7 - c_3 c_6)\psi_7 \psi_8.
\end{aligned}
\tag{B.1}
$$

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] S. P. Timoshenko and J. M. Gere, *Theory of Elastic Stability*, McGraw-Hill, New York, NY, USA, 1961.

[2] G. A. J. Lopez, *Inelastic stability of tapered structural members [Ph.D. thesis]*, University of Minnesota, Minneapolis, Minn, USA, 1998.

[3] AISC-2010, "Specification for structural steel buildings," ANSI/AISC 360-10, American Institute of Steel Construction, Chicago, Ill, USA.

[4] W. F. Chen and E. M. Lui, *Stability Design of Steel Frames*, CRC Press, Boca Raton, Fla, USA, 1992.

[5] S. P. Timoshenko and D. H. Young, *Theory of Structures*, McGraw-Hill, New York, NY, USA, 1965.

[6] K. J. Bathe, *Finite Element Procedures in Engineering Analysis*, Prentice Hall, Englewood, NJ, USA, 1982.

[7] D. L. Karabalis and D. E. Beskos, "Static, dynamic and stability analysis of structures composed of tapered beams," *Computers and Structures*, vol. 16, no. 6, pp. 731–748, 1983.

[8] H. J. Al-Gahtani, "Exact stiffnesses for tapered members," *Journal of Structural Engineering*, vol. 122, no. 10, pp. 1234–1239, 1996.

[9] G. Q. Li and J. J. Li, "A tapered Timoshenko-Euler beam element for analysis of steel portal frames," *Journal of Constructional Steel Research*, vol. 58, no. 12, pp. 1531–1544, 2002.

[10] J. J. Li, G. Q. Li, and S. L. Chan, "A second-order inelastic model for steel frames of tapered members with slender web," *Engineering Structures*, vol. 25, no. 8, pp. 1033–1043, 2003.

[11] J. J. Li and G. Q. Li, *Advanced Analysis and Design of Steel Frames*, John Wiley & Sons, New York, NY, USA, 2007.

[12] S. E. Kim, J. S. Lee, S. H. Choi, and C. S. Kim, "Practical second-order inelastic analysis for steel frames subjected to distributed load," *Engineering Structures*, vol. 26, no. 1, pp. 51–61, 2004.

[13] W. McGuire, R. H. Gallagher, and R. Ziemian, *Matrix Structural Analysis*, John Wiley & Sons, New York, NY, USA, 2nd edition, 1998.

[14] Matlab V7.1, *User's Manual*, MathWorks, Natick, Mass, USA, 2004.

[15] ABAQUS, V.6.10, *Reference Manual*, Dassault Systemes Simulia Corporation, Providence, RI, USA, 2010.

[16] N. M. Rahbari, S. Talatahari, and R. Aalami, "Reliability analysis of semiactive magnetorheological dampers subjected to harmonic excitations," *Advances in Mechanical Engineering*, vol. 2013, Article ID 643927, 7 pages, 2013.

[17] H. Saffari, R. Rahgozar, and R. Jahanshahi, "An efficient method for computation of effective length factor of columns in a steel gabled frame with tapered members," *Journal of Constructional Steel Research*, vol. 64, no. 4, pp. 400–406, 2008.

[18] W. S. King, C. J. Chen, L. Duan, and W. F. Chen, "Plastic analysis of frames with tapered members," *Journal of Architecture and Civil Engineering*, vol. 23, no. 2, pp. 9–19, 2006.

[19] M. P. Saka, "Optimum design of steel frames with tapered members," *Computers and Structures*, vol. 63, no. 4, pp. 797–811, 1997.

[20] SAP2000, Ver.14, *Analysis Reference Manual*, Computers and Structures, Inc., Berkeley, Calif, USA, 2009.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2014, Article ID 471418, 14 pages
http://dx.doi.org/10.1155/2014/471418

Research Article

An Observer-Based Adaptive Iterative Learning Control Using Filtered-FNN Design for Robotic Systems

Ying-Chung Wang and Chiang-Ju Chien

Department of Electronic Engineering, Huafan University, Shihding, New Taipei City 223, Taiwan

Correspondence should be addressed to Ying-Chung Wang; ycwang@cc.hfu.edu.tw

Received 9 August 2013; Accepted 5 November 2013; Published 5 February 2014

Academic Editor: Amir H. Alavi

An observer-based adaptive iterative learning control using a filtered fuzzy neural network is proposed for repetitive tracking control of robotic systems. A state tracking error observer is introduced to design the iterative learning controller using only the measurement of joint position. We first derive an observation error model based on the state tracking error observer. Then, by introducing some auxiliary signals, the iterative learning controller is proposed based on the use of an averaging filter. The main control force consists of a filtered fuzzy neural network used to approximate for unknown system nonlinearity, a robust learning term used to compensate for uncertainty, and a stabilization term used to guarantee the boundedness of internal signals. The adaptive laws combining time domain and iteration domain adaptation are presented to ensure the convergence of learning error. We show that all the adjustable parameters as well as internal signals remain bounded for all iterations. The norm of output tracking error will asymptotically converge to a tunable residual set as iteration goes to infinity.

1. Introduction

Due to the repeatability of operation for robotic systems, it is suitable to perform the control task by using the technique of iterative learning control (ILC) for a repetitive tracking control problem. ILC is basically a non-model-based learning approach which is very effective in dealing with repetitive control tasks [1–5]. Initially, PID-type ILC algorithms which required a certain a priori knowledge of robot dynamics were developed for robot manipulators based on the contraction mapping theory [6–9]. In D-type, PD-type, or PID-type ILC, the acceleration errors of joint variables are required to construct the learning controller. However, the acceleration measurement is unfortunately not realizable in practice and becomes the major disadvantage of D-type ILC. On the other hand, P-type ILC uses only the velocity errors of joint variables for the design of updated learning function. Although the requirement of acceleration measurement is removed, more strict conditions on the robot manipulators are needed for technical analysis. In general, it is hard to apply the traditional PID-type ILC for repetitive tracking control of robot manipulator using only joint position measurement.

Recently, adaptive iterative learning control (AILC) has been widely studied in the research field of the ILC. One of the most attractive advantages of AILC scheme is the capability to deal with the issues of large initial resetting error, large input disturbance, and iteration-varying desired trajectory. In the past decade, the AILC schemes have been utilized for repeated tracking control of robotic systems [10–14], or a class of nonlinear systems [15, 16]. In order to relax the restrict Lipschitz condition on the plant's nonlinearity, the Lyapunov-like approach instead of contraction mapping theory is applied in AILC to analyze the stability and convergence. If the system nonlinearties are unknown, the fuzzy systems or neural networks were often introduced to approximate the nonliearties and provide the basis functions for the design of AILC [17–19]. However, both PID-type ILCs and AILCs require at least the joint velocities to develop the AILC algorithms. If only the measurement of joint positions is available, an observer is one of the possible choice to design the AILC.

In [20], an observer-based ILC for a time-varying nonlinear system was proposed to overcome the unmeasurable states. Although an observer-based ILC system can guarantee that the tracking error will converge to zero, the initial resetting errors at each iteration were not considered. In [21], an observer-based ILC scheme was proposed for a class of nonlinear systems with unknown parametric uncertainties. The Lyapunov-Krasovskii-like composite energy function was applied to analyze the closed-loop stability and learning performance. Nevertheless, the plant nonlinearities must be linearly parameterizable. In [22], a framework for ILC by using an observer to estimate the controlled variable was presented. However, it was necessary to assume that the ILC input converges to a bounded signal. In [23], the observer-based ILC with evolutionary programming algorithm was proposed for MIMO nonlinear systems. The evolutionary programming was applied to search for the optimal and feasible learning gain to speed up the convergence of the ILC and the tracking error will converge to zero via successive learning. But the AILC was developed for MIMO nonlinear plants with nonlinearities satisfying Lipschitz continuous condition. In [24], an observer-based AILC was developed for a class of nonlinear systems with unknown time-varying parameters and unknown time-varying delays. The linear matrix inequality (LMI) approach was used to design the nonlinear state observer. By constructing a Lyapunov-Krasovskii-like composite energy function, the boundedness of the internal signals and the convergence of tracking error can be proved. However, it was assumed that the plant nonlinearities satisfy Lipschitz continuous condition and the unknown system parameters must be linear with respect to some known nonlinear vector-valued functions. In [25], a velocity-observer-based ILC for trajectory tracking of rigid robot manipulators with external disturbances without using the velocity measurement was proposed. However, the inertia matrix of the robotic systems needed to satisfy Lipschitz continuous condition and the initial resetting errors at each iteration were assumed to be zero. Besides, the disturbances were assumed to be repetitive and the velocities were assumed to be bounded.

In order to relax the condition of measurement for joint velocity and without the requirement of Lipschitz condition on the robot unknown nonlinearity and the zero initial resetting errors at each iteration, a new observer-based adaptive iterative learning controller using a new filtered fuzzy neural network (filtered-FNN) is proposed for repetitive tracing control of robotic systems in this paper. A state tracking error observer is firstly presented to deal with the problem that only joint positions are available. Based on this observer, a state tracking error model including a filtered-FNN approximation term can be derived by using an interesting s-domain transfer function technique which is usually utilized in the area of traditional model reference adaptive control [26]. An averaging filter is then proposed to solve the implementation problem of the controller. Under the derived error model, a fuzzy neural learning component is designed to approximate the unknown nonlinearities by a filtered-FNN using state estimation vector as the network input, a robust learning component is constructed to compensate for the uncertainties from approximation error and state estimation error, and a stabilization component is used to guarantee the boundedness of internal signals. The main features of this iterative learning controller and its contributions relative to the related works are summarized as follows.

(1) Compared with most of the works using adaptive iterative learning control for robotic systems or nonlinear systems, this paper can design a realizable adaptive iterative learning controller using only joint position measurement. In other words, it is not necessary to measure the states for the controller design.

(2) A new design approach is introduced to derive the error model so that a filtered-FNN can be applied for compensation of the unknown system nonlinearities. The filtered-FNN can be treated as a dynamic version of traditional FNN which plays an important role in this proposed adaptive iterative learning controller.

(3) Compared with the fuzzy system or neural-network-based adaptive iterative learning controller using state measurement [17–19], the stability analysis becomes more difficult since the boundedness of output tracking error cannot guarantee the boundedness of input signal. A new analysis to prove the regularity of internal signals is successfully derived in this paper so that we ensure the boundedness of all the adjustable parameters and internal signals during the learning process. Furthermore, we guarantee that the norm of output tracking error will asymptotically converge to a tunable residual set which depends on the design parameters if iteration number is large enough.

This paper is organized as follows. In Section 2, a problem formulation is given. The error model between system output and desired output is derived in Section 3. Based on the derived error model, the filtered-FNN-based AILC using observer design is presented in Section 4. Analysis of closed-loop stability and learning performance will be studied extensively in Section 5. Finally, a conclusion is made in Section 6. The detailed description of the proposed filtered FNN is given in the Appendix.

In the subsequent discussions, the following notations will be used.

(1) $|\cdot|$ denotes the absolute value of a scalar or the Euclidean or any other consistent norm of a vector or matrix.

(2) $L_{pe}[0,T]$ denotes the set of Lebesgue measurable (or piecewise continuous) real-valued (vector) functions with

$$|\cdot|_{pe} = \begin{cases} \left(\int_0^T |\cdot|^p dt\right)^{1/p}, & \text{if } p \in [1,\infty), \\ \sup\limits_{0\le t\le T} |\cdot|, & \text{if } p = \infty. \end{cases} \tag{1}$$

(3) $\|(\cdot)_t\|_\infty = \sup_{\tau\le t}|(\cdot)(\tau)|$ denotes the truncated L_∞ norm of the argument function or vector [26].

(4) $\|G(s)\|_\infty$ denotes the H_∞ norm of the transfer function $G(s)$.

(5) $G(s)[u(t)]$ denotes the filtered version of $u(t)$ with any proper or strictly proper transfer function $G(s)$.

2. Problem Formulation

In this paper, we consider an uncertain robotic system with n rigid bodies which can perform a given task repeatedly over a finite time interval $[0,T]$ as follows:

$$D\left(q^j(t)\right)\ddot{q}^j(t) + B\left(q^j(t),\dot{q}^j(t)\right)\dot{q}^j(t) + F\left(q^j(t),\dot{q}^j(t)\right) = u^j(t), \tag{2}$$

where $j \in Z_+$ denotes the index of iteration number and $t \in [0,T]$ denotes the time index. The signals $q^j(t)$, $\dot{q}^j(t)$, $\ddot{q}^j(t) \in \mathscr{R}^{n\times 1}$ are, respectively, the generalized joint position, joint velocity, and joint acceleration vectors. $D(q^j(t)) \in \mathscr{R}^{n\times n}$ is the inertia matrix, $B(q^j(t), \dot{q}^j(t)) \in \mathscr{R}^{n\times n}$ is the centripetal plus Coriolis force matrix, $F(q^j(t), \dot{q}^j(t)) \in \mathscr{R}^{n\times 1}$ are the gravitational plus frictional forces, and $u^j \in \mathscr{R}^{n\times 1}$ is the joint torque vector. It is noted the inertia matrix $D(q^j(t))$ is assumed to be positive definite and bounded for all $t \in [0,T]$ and iteration $j \geq 1$ as

$$0 < m_1 I_{n\times n} \leq D\left(q^j(t)\right) \leq m_2 I_{n\times n}, \tag{3}$$

where $m_1, m_2 > 0$ and $I_{n\times n}$ is an $n \times n$ identity matrix. Since the inverse of inertia matrix exists for all joint variables, the dynamic formulation of the robotic system can be written as follows:

$$\begin{aligned}\ddot{q}^j(t) &= -D^{-1}\left(q^j(t)\right)\\ &\quad\times\left[B\left(q^j(t),\dot{q}^j(t)\right)\dot{q}^j(t) + F\left(q^j(t),\dot{q}^j(t)\right)\right]\\ &\quad + D^{-1}\left(q^j(t)\right)u^j(t).\end{aligned} \tag{4}$$

Let $f(q^j(t),\dot{q}^j(t)) = D^{-1}(q^j(t))[B(q^j(t),\dot{q}^j(t))\dot{q}^j(t) + F(q^j(t),\dot{q}^j(t))] \in \mathscr{R}^{n\times 1}$ and $b(q^j(t)) = D^{-1}(q^j(t)) \in \mathscr{R}^{n\times n}$ and choose the output variable as $y^j(t) = q^j(t) \in \mathscr{R}^{n\times 1}$, state variable as $X^j(t) = [x_1^j(t)^\top, x_2^j(t)^\top]^\top = [y^j(t)^\top, \dot{y}^j(t)^\top]^\top = [q^j(t)^\top, \dot{q}^j(t)^\top]^\top \in \mathscr{R}^{2n\times 1}$, then we have

$$\begin{aligned}\dot{x}_1^j(t) &= x_2^j(t),\\ \dot{x}_2^j(t) &= -f\left(X^j(t)\right) + b\left(X^j(t)\right)u^j(t),\\ y^j(t) &= x_1^j(t),\end{aligned} \tag{5}$$

or equivalently in the following state-space form:

$$\begin{aligned}\dot{X}^j(t) &= AX^j(t) + B\left(-f\left(X^j(t)\right) + b\left(X^j(t)\right)u^j(t)\right),\\ y^j(t) &= C^\top X^j(t),\end{aligned} \tag{6}$$

where

$$\begin{gathered}A = \begin{bmatrix}0 & I_{n\times n}\\ 0 & 0\end{bmatrix} \in \mathscr{R}^{2n\times 2n}, \qquad B = \begin{bmatrix}0\\ I_{n\times n}\end{bmatrix} \in \mathscr{R}^{2n\times n},\\ C = \begin{bmatrix}I_{n\times n}\\ 0\end{bmatrix} \in \mathscr{R}^{2n\times n},\end{gathered} \tag{7}$$

Here, $0 < b(X^j(t)) \leq m_1^{-1} I_{n\times n}$. In this paper, we assume that only joint position $y^j(t) = q^j(t)$ is measurable for controller design. Let the signals $q_d(t)$, $\dot{q}_d(t)$, $\ddot{q}_d(t) \in \mathscr{R}^{n\times 1}$ be, respectively, the desired generalized joint position, joint velocity, and joint acceleration vectors; the desired state trajectory can be defined as $X_d(t) = [x_{d_1}(t)^\top, x_{d_2}(t)^\top]^\top = [y_d(t)^\top, \dot{y}_d(t)^\top]^\top = [q_d(t)^\top, \dot{q}_d(t)^\top]^\top$. Now, given a specified desired output trajectory $y_d(t)$, $t \in [0,T]$ and an initial desired output $y_d(0) \neq y^j(0)$ for all $j \geq 1$, the control objective for the robotic systems executing a repeatable task is to force the output $y^j(t)$ to follow $y_d(t)$ as close as possible.

3. Derivations of Error Model and Controller

3.1. Derive the Error Model. Define an output tracking error as $e^j(t) = y_d(t) - y^j(t)$ and state tracking errors as $e_1^j(t) = y_d(t) - y^j(t)$, $e_2^j(t) = \dot{y}_d(t) - \dot{y}^j(t)$. It is assumed that the initial output tracking error vector at each iteration is not necessarily zero, small, and fixed but satisfies $|e^j(0)| = \varepsilon^j$ for a known positive constants ε^j since the joint position vector $y^j(t)$ is measurable. Let the state tracking error vector be defined as

$$\begin{aligned}E^j(t) &= X_d(t) - X^j(t) = \left[e_1^j(t)^\top, e_2^j(t)^\top\right]^\top\\ &= \left[e^j(t)^\top, \dot{e}^j(t)^\top\right]^\top \in \mathscr{R}^{2n\times 1}.\end{aligned} \tag{8}$$

Then we can derive $\ddot{e}^j(t)$ as follows:

$$\begin{aligned}\ddot{e}^j(t) &= \ddot{y}_d(t) + f\left(X^j(t)\right) - b\left(X^j(t)\right)u^j(t)\\ &= -K_c^\top E^j(t) + f\left(X^j(t)\right) + \ddot{y}_d(t) + K_c^\top E^j(t)\\ &\quad - b\left(X^j(t)\right)u^j(t),\end{aligned} \tag{9}$$

where $K_c = [k_2^c I, k_1^c I]^\top \in \mathscr{R}^{2n\times n}$ is a feedback gain matrix such that the characteristic polynomial of $A_c = A - BK_c^\top$ is Hurwitz. Therefore, the tracking error dynamics will satisfy

$$\begin{aligned}&\dot{E}^j(t)\\ &= A_c E^j(t)\\ &\quad + B\left[f\left(X^j(t)\right) + \ddot{y}_d(t)\right.\\ &\qquad \left. + K_c^\top E^j(t) - b\left(X^j(t)\right)u^j(t)\right]\end{aligned}$$

$$
\begin{aligned}
&= A_c E^j(t) \\
&\quad + B\left[f\left(X^j(t)\right) + \ddot{y}_d(t) + K_c^\top X_d(t)\right. \\
&\qquad \left. - K_c^\top X^j(t) - u^j(t) + \left(1 - b\left(X^j(t)\right)\right) u^j(t)\right], \\
&= A_c E^j(t) \\
&\quad + B\left[h\left(X^j(t)\right) - u^j(t) + \left(1 - b\left(X^j(t)\right)\right) u^j(t)\right], \\
&\qquad e^j(t) = C^\top E^j(t),
\end{aligned} \tag{10}
$$

where $h(X^j(t)) \equiv f(X^j(t)) - K_c^\top X^j(t) + \ddot{y}_d(t) + K_c^\top X_d(t)$. Note that the state tracking error vector $E^j(t)$ in the tracking error dynamics (10) is not assumed to be available for measurement. Hence, it is necessary to construct a state estimation vector $\widehat{X}^j(t) = [\widehat{x}_1^j(t)^\top, \widehat{x}_2^j(t)^\top]^\top = [\widehat{y}^j(t)^\top, \dot{\widehat{y}}^j(t)^\top]^\top \in \mathscr{R}^{2n\times 1}$ for estimation of the state vector. In order to construct a state estimation vector, we first define an output tracking error estimation vector as $\widehat{e}^j(t) = y_d(t) - \widehat{y}^j(t)$. Then the state tracking error estimation vector can be defined as

$$
\begin{aligned}
\widehat{E}^j(t) &= X_d(t) - \widehat{X}^j(t) = \left[\widehat{e}_1^j(t)^\top, \widehat{e}_2^j(t)^\top\right]^\top \\
&= \left[\widehat{e}^j(t)^\top, \dot{\widehat{e}}^j(t)^\top\right]^\top \in \mathscr{R}^{2n\times 1}.
\end{aligned} \tag{11}
$$

The state tracking error observer is designed as

$$
\begin{aligned}
&\dot{\widehat{E}}^j(t) = A_c \widehat{E}^j(t) + K_o\left(e^j(t) - \widehat{e}^j(t)\right), \quad \widehat{E}^j(0) = 0, \\
&\qquad \widehat{e}^j(t) = C^\top \widehat{E}^j(t),
\end{aligned} \tag{12}
$$

where $K_o = [k_1^o I, k_2^o I]^\top \in \mathscr{R}^{2n\times n}$ is the observer gain vector such that the characteristic polynomial of $A_o = A_c - K_o C^\top$ is Hurwitz. Define an output observation error vector as $\widetilde{e}^j(t) = e^j(t) - \widehat{e}^j(t)$. Then the state observation error vector can be defined as

$$
\begin{aligned}
\widetilde{E}^j(t) &= E^j(t) - \widehat{E}^j(t) = \left[\widetilde{e}_1^j(t)^\top, \widetilde{e}_2^j(t)^\top\right]^\top \\
&= \left[e^j(t)^\top - \widehat{e}^j(t)^\top, \dot{e}^j(t)^\top - \dot{\widehat{e}}^j(t)^\top\right]^\top \in \mathscr{R}^{2n\times 1}.
\end{aligned} \tag{13}
$$

By using (10) and (12), we have the following observation error dynamics:

$$
\begin{aligned}
&\dot{\widetilde{E}}^j(t) = A_o \widetilde{E}^j(t) \\
&\quad + B\left[h\left(X^j(t)\right) - u^j(t) + \left(1 - b\left(X^j(t)\right)\right) u^j(t)\right], \\
&\qquad \widetilde{e}^j(t) = C^\top \widetilde{E}^j(t).
\end{aligned} \tag{14}
$$

Note that $|\widetilde{e}^j(0)| = |e^j(0) - \widehat{e}^j(0)| = |e^j(0)| = \varepsilon^j$. Based on the universal approximation theorem, we know that the nonlinear function $h(X^j(t))$ can be approximated by a traditional FNN [19] $W^j(t)^\top O^{(3)}(X^j(t))$. Here $O^{(3)}(X^j(t)) \in \mathscr{R}^{M\times 1}$ is the basis function vector with M being the number of rule nodes and $W^j(t) \in \mathscr{R}^{M\times n}$ is the weight matrix of the output layer. According to the universal approximation theorem, there will exist an optimal weight matrix W^* such that $h(X^j(t)) = W^{*\top} O^{(3)}(X^j(t)) + \epsilon(X^j(t))$, where $\epsilon(X^j(t))$ is the approximation error satisfying $|\epsilon(X^j(t))| \le \epsilon^*$ in a certain compact set. This implies that (14) can be rewritten as

$$
\begin{aligned}
&\dot{\widetilde{E}}^j(t) = A_o \widetilde{E}^j(t) \\
&\quad + B\left[W^{*\top} O^{(3)}\left(X^j(t)\right) + \epsilon^j\left(X^j(t)\right) + \ddot{y}_d(t)\right. \\
&\qquad \left. + K_c^\top X_d(t) - u^j(t) + \left(1 - b\left(X^j(t)\right)\right) u^j(t)\right] \\
&= A_o \widetilde{E}^j(t) \\
&\quad + B\left[W^{*\top} O^{(3)}\left(\widehat{X}^j(t)\right) - u^j(t)\right. \\
&\qquad \left. + \delta^j(t) + \left(1 - b\left(X^j(t)\right)\right) u^j(t)\right], \\
&\qquad \widetilde{e}^j(t) = C^\top \widetilde{E}^j(t),
\end{aligned} \tag{15}
$$

where $\delta^j(t) = W^{*\top}(O^{(3)}(X^j(t)) - O^{(3)}(\widehat{X}^j(t))) + \epsilon(X^j(t))$ is the lumped uncertainty term which includes the difference between FNN networks with input using state $X^j(t)$ and estimated state $\widehat{X}^j(t)$ and the network approximation error. It is easily proven that the lumped uncertainty term satisfies $|\delta^j(t)| \le \delta^*$. However, it is noted that the value of the unknown constant δ^* might be large.

To see how to design the iterative learning controller $u^j(t)$ to achieve the control objective, we now adopt the mixed use of a time signal and a Laplace transfer function to obtain the explicit expression of $\widetilde{e}^j(t)$ in (15) in time domain with a filtered version as follows:

$$
\begin{aligned}
\widetilde{e}^j(t) = H(s)&\left[W^{*\top} O^{(3)}\left(\widehat{X}^j(t)\right) - u^j(t) + \delta^j(t)\right. \\
&\left. + \left(1 - b\left(X^j(t)\right)\right) u^j(t)\right],
\end{aligned} \tag{16}
$$

where $H(s) = C^\top(sI - A_o)^{-1}B = (1/(s^2 + (k_1^c + k_1^o)s + (k_1^c k_1^o + k_2^c k_2^o)))I_{n\times n}$. The observer gain vector $K_o = [k_1^o I_{n\times n}, k_2^o I_{n\times n}]^\top$ is chosen such that $H(s) = (1/\ell(s)L(s))I_{n\times n}$ with $\ell(s) = s + \lambda$ and $L(s)$ being any

Hurwitz polynomial of order 1. Then (16) can be rewritten as

$$\begin{aligned}\tilde{e}^j(t) &= H(s)L(s)\left[W^{*\top}\frac{1}{L(s)}\left[O^{(3)}\left(\hat{X}^j(t)\right)\right]\right.\\ &\quad -\frac{1}{L(s)}\left[u^j(t)\right]\\ &\quad \left.+\frac{1}{L(s)}\left[\left(1-b\left(X^j(t)\right)\right)u^j(t)+\delta^j(t)\right]\right]\\ &= \frac{1}{\ell(s)}\left[W^{*\top}\frac{1}{L(s)}\left[O^{(3)}\left(\hat{X}^j(t)\right)\right]\right.\\ &\quad \left.-\frac{1}{L(s)}\left[u^j(t)\right]+\delta_L^j(t)\right],\end{aligned} \tag{17}$$

where $\delta_L^j(t) = (1/L(s))[(1-b(X^j(t)))u^j(t)+\delta^j(t)]$.

3.2. Construct Some Useful Signals. According to the filtered version of output tracking error model (17), we define an augmented signal with filtered version as

$$y_a^j(t) = \frac{1}{\ell(s)}\left[v^j(t)-\frac{1}{L(s)}\left[u^j(t)\right]\right], \quad y_a^j(0)=0, \tag{18}$$

where $v^j(t)$ is an auxiliary input to be designed later. Then, design an auxiliary error signal as

$$e_a^j(t) = \tilde{e}^j(t) - y_a^j(t). \tag{19}$$

The initial condition $e_a^j(0)$ will satisfy $|e_a^j(0)| = |\tilde{e}^j(0) - y_a^j(0)| = |\tilde{e}^j(0)| = |e^j(0)| \equiv \varepsilon^j$. Substituting (17) and (18) into (19), we can find that

$$\begin{aligned}e_a^j(t) &= \tilde{e}^j(t) - y_a^j(t)\\ &= \frac{1}{\ell(s)}\left[W^{*\top}\frac{1}{L(s)}\left[O^{(3)}\left(\hat{X}^j(t)\right)\right]-v^j(t)+\delta_L^j(t)\right].\end{aligned} \tag{20}$$

It is noted that we apply the Laplace operation to derive the tracking error model (20) for technical analysis later as that used in the traditional model reference adaptive control [26]. The equivalent time-domain state space representation of (20) is

$$\begin{aligned}\dot{e}_a^j(t) &= -\lambda e_a^j(t) + W^{*\top}\frac{1}{L(s)}\left[O^{(3)}\left(\hat{X}^j(t)\right)\right]\\ &\quad -v^j(t)+\delta_L^j(t).\end{aligned} \tag{21}$$

In order to overcome the uncertainty from initial output tracking error, a dead-zone signal $e_\phi^j(t)$ is introduced as follows:

$$e_\phi^j(t) = e_a^j(t) - \phi^j(t)\,\mathrm{sat}\left(\frac{e_a^j(t)}{\phi^j(t)}\right), \quad \phi^j(t) = \varepsilon^j e^{-\lambda t}, \tag{22}$$

where $\mathrm{sat}(e_a^j(t)/\phi^j(t)) = [\mathrm{sat}(e_{a,1}^j(t)/\phi^j(t)),\ldots,(e_{a,n}^j(t)/\phi^j(t))]^\top \in \mathscr{R}^{n\times 1}$ and

$$\mathrm{sat}\left(\frac{e_{a,i}^j(t)}{\phi^j(t)}\right) = \begin{cases}1, & \text{if } e_{a,i}^j(t) > \phi^j(t),\\ \dfrac{e_{a,i}^j(t)}{\phi^j(t)}, & \text{if } \left|e_{a,i}^j(t)\right| \le \phi^j(t), \quad i=1,\ldots,n,\\ -1, & \text{if } e_{a,i}^j(t) < -\phi^j(t),\end{cases} \tag{23}$$

and $\phi^j(t)$ is the width of boundary layer. Note that $\phi^j(t)$ is designed to decrease along time axis with the initial condition chosen as $\phi^j(0) = \varepsilon^j$ for jth iteration and $0 < \varepsilon^j e^{-\lambda T} \le \phi^j(t) \le \varepsilon^j$, $\forall t \in [0,T]$, $j \ge 1$. According to (22), it is easy to show that $e_\phi^j(0) = 0$, $\forall j \ge 1$. Now let us differentiate $(1/2)e_\phi^j(t)^\top e_\phi^j(t)$ as follows:

$$\begin{aligned}&\frac{1}{2}\frac{d}{dt}e_\phi^j(t)^\top e_\phi^j(t)\\ &\quad = e_\phi^j(t)^\top \dot{e}_\phi^j(t)\\ &\quad = e_\phi^j(t)^\top\left(\dot{e}_a^j(t) - \mathrm{sgn}\left(e_\phi^j(t)\right)\dot{\phi}^j(t)\right)\\ &\quad = e_\phi^j(t)^\top\left\{-\lambda e_a^j(t) - \mathrm{sgn}\left(e_\phi^j(t)\right)\dot{\phi}^j(t)\right\}\\ &\qquad + e_\phi^j(t)^\top\left\{W^{*\top}\frac{1}{L(s)}\left[O^{(3)}\left(\hat{X}^j(t)\right)\right]-v^j(t)+\delta_L^j(t)\right\}\\ &\quad = -\lambda e_\phi^j(t)^\top e_\phi^j(t)\\ &\qquad + e_\phi^j(t)^\top\left\{W^{*\top}\frac{1}{L(s)}\left[O^{(3)}\left(\hat{X}^j(t)\right)\right]-v^j(t)+\delta_L^j(t)\right\},\end{aligned} \tag{24}$$

where $\mathrm{sgn}(e_\phi^j(t)) = [\mathrm{sgn}(e_{\phi,1}^j(t)),\ldots,\mathrm{sgn}(e_{\phi,n}^j(t))]^\top$ is the typical signum function vector. In the following, we will show that the uncertainty term $\delta_L^j(t)$ in (17) can be bounded in a linearly parameterized form if the following normalization signal $m^j(t)$ [27] is utilized

$$m^j(t) = \frac{\delta_2}{s+\delta_1}\left[1+\left|u^j(t)\right|\right], \quad m^j(0) > \frac{\delta_2}{\delta_1}, \tag{25}$$

where $\delta_1, \delta_2 > 0$ and $\delta_1 < \delta_1^*$. Here δ_1^* is the least positive constant such that $1/L(s-\delta_1^*)$ is a stable system. In practical, δ_1 can be chosen as small as possible. Note that $1-b(X^j(t))$ and $\delta^j(t)$ are bounded and $1/L(s)$ is a strictly proper stable transfer function. Hence, according to the definition of $\delta_L(t)$ and by using Lemma 3.1 in [27], we can prove that

$$\begin{aligned}\left|\delta_L^j(t)\right| &\le \left|\frac{1}{L(s)}\left[\left(1-b\left(X^j(t)\right)\right)u^j(t)+\delta^j(t)\right]\right|\\ &\le \theta_1^* m^j(t) + \theta_2^* = [\theta_1^*, \theta_2^*]\begin{bmatrix}m^j(t)\\ 1\end{bmatrix}\\ &\equiv \theta^{*\top}Y^j(t)\end{aligned} \tag{26}$$

for some positive constants θ_1^*, θ_2^*.

3.3. Design the Filtered-FNN-Based Iterative Learning Controller. Based on the aforementioned derived error model and the useful signals, we now design $u^j(t)$ and $v^j(t)$ as follows:

$$u^j(t) = \frac{L(s)}{F(\tau s)}\left[v^j(t)\right], \tag{27}$$

$$v^j(t) = W^j(t)^\top O^{(4)}\left(\widehat{X}^j(t)\right) + \operatorname{sat}\left(\frac{e_a^j(t)}{\phi^j(t)}\right)\theta^j(t)^\top Y^j(t) + e_\phi^j(t)\,Y^j(t)^\top Y^j(t). \tag{28}$$

Here $W^j(t)^\top O^{(4)}(\widehat{X}^j(t))$ is a filtered-FNN (see Appendix) used as an approximator to compensate for $h(X^j(t))$, $\operatorname{sat}(e_a^j(t)/\phi^j(t))\theta^j(t)^\top Y^j(t)$ is a robust learning term used to overcome the uncertainties due to function approximation error and the error induced by using state estimation, and $e_\phi^j(t)Y^j(t)^\top Y^j(t)$ is a stabilization term used to guarantee the boundedness of all the closed-loop signals. In this controller, $W^j(t)$ is the weight matrix of the filtered-FNN and $\theta^j(t)$ is control parameter vector which are used to compensate for the unknown W^* and θ^*, respectively. Furthermore, $F(\tau s) = (\tau s + 1)^2$ with $\tau > 0$ being a small constant. In the literature, $1/F(\tau s)$ is referred to as an *averaging filter*, which is obviously a low-pass filter whose bandwidth can be arbitrarily enlarged as τ approaches 0. If we define the parameter error as $\widetilde{W}(t) = W^j(t) - W^*$ and $\widetilde{\theta}^j(t) = \theta^j(t) - \theta^*$ and substitute (28) into (24), we have the following error dynamics for technical analysis later:

$$\begin{aligned}
&\frac{1}{2}\frac{d}{dt}e_\phi^j(t)^\top e_\phi^j(t) \\
&\quad = -\lambda e_\phi^j(t)^\top e_\phi^j(t) \\
&\qquad + e_\phi^j(t)^\top\Bigg\{-\widetilde{W}^j(t)^\top O^{(4)}\left(\widehat{X}^j(t)\right) \\
&\qquad\qquad - \operatorname{sat}\left(\frac{e_a^j(t)}{\phi^j(t)}\right)\theta^j(t)^\top Y^j(t) \\
&\qquad\qquad - e_\phi^j(t)\,Y^j(t)^\top Y^j(t) + \delta_L^j(t)\Bigg\} \\
&\quad \le -\lambda e_\phi^j(t)^\top e_\phi^j(t) - e_\phi^j(t)^\top\widetilde{W}^j(t)^\top O^{(4)}\left(\widehat{X}^j(t)\right) \\
&\qquad - \left|e_\phi^j(t)\right|\widetilde{\theta}^j(t)^\top Y^j(t) - e_\phi^j(t)^\top e_\phi^j(t)\,Y^j(t)^\top Y^j(t).
\end{aligned} \tag{29}$$

A set of stable adaptive laws is necessary to tune the control parameters. The adaptive laws combining time domain and iteration domain adaptation without knowledge of known bounds on optimal parameters or dead zone mechanism are proposed as follows:

$$\begin{aligned}
(1-\gamma_1)\dot{W}^j &= -\gamma_1 W^j + \gamma_1 W^{j-1}(t) \\
&\quad + \beta_1 O^{(4)}\left(\widehat{X}^j(t)\right)e_\phi^j(t)^\top,
\end{aligned} \tag{30}$$

$$(1-\gamma_2)\dot{\theta}^j = -\gamma_2\theta^j(t) + \gamma_2\theta^{j-1}(t) + \beta_2\left|e_\phi^j(t)\right|Y^j(t) \tag{31}$$

with $W^j(0) = W^{j-1}(T), \theta^j(0) = \theta^{j-1}(T)$ for $j \ge 1$, and $0 < \gamma_1, \gamma_2 < 1$, $\beta_1, \beta_2 > 0$. In these adaptive laws, γ_1, γ_2 and β_1, β_2 are defined as the weighting gains and adaptation gains, respectively. For the first iteration, we set $W^0(t) = W^0$ and $\theta^0(t) = \theta^0$ to be any constant matrix or vector, $\forall t \in [0, T]$ and $\forall j \ge 1$. Equations (30) and (31) will become pure time-domain adaptive laws if $\gamma_1 = \gamma_2 = 0$, or pure iteration-domain adaptive laws if $\gamma_1 = \gamma_2 = 1$.

4. Analysis of Stability and Convergence

To prove the stability and convergence of the proposed learning system, we first give the following three lemmas.

Lemma 1. *Consider the robotic system* (2) *performing a repetitive control task. If one applies the observer-based adaptive filtered-FNN iterative learning controller* (18), (19), (22), (25), (27), *and* (28) *with adaptation laws* (30) *and* (31), *then one guarantees that* $e_\phi^1(t), e_a^1(t), \widetilde{W}^1(t)$, *and* $\widetilde{\theta}^1(t)$ *are bounded.*

Proof. Choose a Lyapunov-like positive function as

$$\begin{aligned}
V_a^j(t) &= \frac{1}{2}e_\phi^j(t)^\top e_\phi^j(t) + \frac{(1-\gamma_1)}{2\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(t)^\top\widetilde{W}^j(t)\right\} \\
&\quad + \frac{(1-\gamma_2)}{2\beta_2}\widetilde{\theta}^j(t)^\top\widetilde{\theta}^j(t)
\end{aligned} \tag{32}$$

and compute its derivative with respect to time t along (29), (30), and (31); then we have

$$\begin{aligned}
\dot{V}_a^j(t) &= \frac{1}{2}\frac{d}{dt}e_\phi^j(t)^\top e_\phi^j(t) + \frac{(1-\gamma_1)}{\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(t)^\top\dot{\widetilde{W}}^j(t)\right\} \\
&\quad + \frac{(1-\gamma_2)}{\beta_2}\widetilde{\theta}^j(t)^\top\dot{\widetilde{\theta}}^j(t) \\
&\le -\lambda e_\phi^j(t)^\top e_\phi^j(t) + e_\phi^j(t)^\top\widetilde{W}^j(t)^\top O^{(4)}\left(\widehat{X}^j(t)\right) \\
&\quad - \left|e_\phi^j(t)\right|\widetilde{\theta}^j(t)^\top Y^j(t) \\
&\quad + \frac{1}{\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(t)^\top(1-\gamma_1)\dot{\widetilde{W}}^j(t)\right\} \\
&\quad + \frac{1}{\beta_2}\widetilde{\theta}^j(t)^\top(1-\gamma_2)\dot{\widetilde{\theta}}^j(t).
\end{aligned} \tag{33}$$

Since $-\gamma_1 W^j(t) + \gamma_1 W^{j-1}(t) = -\gamma_1 \widetilde{W}^j(t) + \gamma_1 \widetilde{W}^{j-1}(t)$ and $-\gamma_2 \theta^j(t) + \gamma_2 \theta^{j-1}(t) = -\gamma_2 \widetilde{\theta}^j(t) + \gamma_2 \widetilde{\theta}^{j-1}(t)$, $V_a^j(t)$ in (33) can be simplified by using (30) and (31) as

$$\begin{aligned}
\dot{V}_a^j(t) &\le -\lambda e_\phi^j(t)^\top e_\phi^j(t) - \frac{\gamma_1}{\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(t)^\top \widetilde{W}^j(t)\right\} \\
&\quad + \frac{\gamma_1}{\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(t)^\top \widetilde{W}^{j-1}(t)\right\} \\
&\quad - \frac{\gamma_2}{\beta_2}\widetilde{\theta}^j(t)^\top \widetilde{\theta}^j(t) + \frac{\gamma_2}{\beta_2}\widetilde{\theta}^j(t)^\top \widetilde{\theta}^{j-1}(t) \\
&= -\lambda e_\phi^j(t)^\top e_\phi^j(t) \\
&\quad - \frac{\gamma_1}{2\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(t)^\top \widetilde{W}^j(t)\right\} - \frac{\gamma_2}{2\beta_2}\widetilde{\theta}^j(t)^\top \widetilde{\theta}^j(t) \\
&\quad - \frac{\gamma_1}{2\beta_1}\operatorname{tr}\left\{\left(\widetilde{W}^j(t) - \widetilde{W}^{j-1}(t)\right)^\top \left(\widetilde{W}^j(t) - \widetilde{W}^{j-1}(t)\right)\right\} \\
&\quad - \frac{\gamma_2}{2\beta_2}\left(\widetilde{\theta}^j(t) - \widetilde{\theta}^{j-1}(t)\right)^\top \left(\widetilde{\theta}^j(t) - \widetilde{\theta}^{j-1}(t)\right) \\
&\quad + \frac{\gamma_1}{2\beta_1}\operatorname{tr}\left\{\widetilde{W}^{j-1}(t)^\top \widetilde{W}^{j-1}(t)\right\} + \frac{\gamma_2}{2\beta_2}\widetilde{\theta}^{j-1}(t)^\top \widetilde{\theta}^{j-1}(t) \\
&\le \frac{\gamma_1}{2\beta_1}\operatorname{tr}\left\{\widetilde{W}^{j-1}(t)^\top \widetilde{W}^{j-1}(t)\right\} + \frac{\gamma_2}{2\beta_2}\widetilde{\theta}^{j-1}(t)^\top \widetilde{\theta}^{j-1}(t) \\
&\equiv V_b^{j-1}(t),
\end{aligned} \tag{34}$$

where we use the property of $\operatorname{tr}\{\widetilde{W}^j(t)^\top O^{(4)}(\widehat{X}^j(t)) e_\phi^j(t)^\top\} = e_\phi^j(t)^\top \widetilde{W}^j(t)^\top O^{(4)}(\widehat{X}^j(t))$. Since $\widetilde{W}^0(t) = W^0(t) - W^* = W^0 - W^* \equiv \overline{W}^0$ and $\widetilde{\theta}^0(t) = \theta^0(t) - \theta^* = \theta^0 - \theta^* \equiv \overline{\theta}^0$ are bounded for all $t \in [0, T]$ so that if $j = 1$, (34) can be rewritten as

$$\dot{V}_a^1(t) \le V_b^0(t) = \frac{\gamma_1}{2\beta_1}\operatorname{tr}\left\{\overline{W}^{0\top}\overline{W}^0\right\} + \frac{\gamma_2}{2\beta_2}\overline{\theta}^{0\top}\overline{\theta}^0. \tag{35}$$

Note that the initial value $V_a^1(0)$ is bounded since $e_\phi^1(0) = 0$, $\widetilde{W}^1(0) = W^1(0) - W^* = W^0(T) - W^* = \overline{W}^0$ and $\widetilde{\theta}^1(0) = \theta^1(0) - \theta^* = \theta^0(T) - \theta^* = \overline{\theta}^0$. Together with the result of (35), it readily implies $V_a^1(t), e_\phi^1(t), \widetilde{W}^1(t), \widetilde{\theta}^1(t) \in L_{\infty e}[0, T]$. Since the filtered basis function vector $O^{(4)}(\widehat{X}^j(t))$ is bounded for all $j \ge 1$, we conclude that $e_a^1(t)$ (by (22)) $\in L_{\infty e}[0, T]$. □

Lemma 2. *Consider the problem setup in Lemma 1. The proposed observer-based adaptive filtered-FNN iterative learning controller guarantees that $e_\phi^j(T), \widetilde{W}^j(T)$, and $\widetilde{\theta}^j(T)$ are bounded, for all $j \ge 1$ as well as $\lim_{j\to\infty}\int_0^T e_\phi^j(t)^\top e_\phi^j(t)dt = 0$ and $\lim_{j\to\infty} e_\phi^j(T)^\top e_\phi^j(T) = 0$.*

Proof. Define a positive function $V^j(T)$ as

$$\begin{aligned}
V^j(T) &= \int_0^T \left[\frac{\gamma_1}{2\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(t)^\top \widetilde{W}^j(t)\right\} + \frac{\gamma_2}{2\beta_2}\widetilde{\theta}^j(t)^\top \theta^j(t)\right] dt \\
&\quad + \frac{1-\gamma_1}{2\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(T)^\top \widetilde{W}^j(T)\right\} \\
&\quad + \frac{1-\gamma_2}{2\beta_2}\widetilde{\theta}^j(T)^\top \theta^j(T).
\end{aligned} \tag{36}$$

Using the technique of integration by parts, we have

$$\begin{aligned}
&\frac{(1-\gamma_1)}{2\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(T)^\top \widetilde{W}^j(T)\right\} \\
&\quad = \frac{(1-\gamma_1)}{\beta_1}\int_0^T \operatorname{tr}\left\{\widetilde{W}^j(t)^\top \dot{\widetilde{W}}^j(t)\right\} dt \\
&\qquad + \frac{(1-\gamma_1)}{2\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(0)^\top \widetilde{W}^j(0)\right\}, \\
&\frac{(1-\gamma_2)}{2\beta_2}\widetilde{\theta}^j(T)^\top \theta^j(T) \\
&\quad = \frac{(1-\gamma_2)}{\beta_2}\int_0^T \widetilde{\theta}^j(t)^\top \dot{\widetilde{\theta}}^j(t)\, dt \\
&\qquad + \frac{(1-\gamma_2)}{2\beta_2}\widetilde{\theta}^j(0)^\top \widetilde{\theta}^j(0).
\end{aligned} \tag{37}$$

The difference between $V^j(T)$ and $V^{j-1}(T)$ can be derived by the facts of $\widetilde{W}^j(0) = \widetilde{W}^{j-1}(T)$ and $\widetilde{\theta}^j(0) = \widetilde{\theta}^{j-1}(T)$ as follows:

$$\begin{aligned}
V^j(T) - V^{j-1}(T) &= \int_0^T \left[\frac{\gamma_1}{2\beta_1}\left(\operatorname{tr}\left\{\widetilde{W}^j(t)^\top \widetilde{W}^j(t)\right\} - \operatorname{tr}\left\{\widetilde{W}^{j-1}(t)^\top \widetilde{W}^{j-1}(t)\right\}\right)\right. \\
&\qquad \left. + \frac{\gamma_2}{2\beta_2}\left(\left(\widetilde{\theta}^j(t)^\top \widetilde{\theta}^j(t)\right) - \left(\widetilde{\theta}^{j-1}(t)^\top \widetilde{\theta}^{j-1}(t)\right)\right)\right] dt \\
&\quad + \frac{(1-\gamma_1)}{\beta_1}\int_0^T \operatorname{tr}\left\{\widetilde{W}^j(t)^\top \dot{\widetilde{W}}^j(t)\right\} dt \\
&\quad + \frac{(1-\gamma_1)}{2\beta_1}\operatorname{tr}\left\{\widetilde{W}^j(0)^\top \widetilde{W}^j(0)\right\} \\
&\quad - \frac{(1-\gamma_1)}{2\beta_1}\operatorname{tr}\left\{\widetilde{W}^{j-1}(T)^\top \widetilde{W}^{j-1}(T)\right\} \\
&\quad + \frac{(1-\gamma_2)}{\beta_2}\int_0^T \widetilde{\theta}^j(t)^\top \dot{\widetilde{\theta}}^j(t)\, dt + \frac{(1-\gamma_2)}{2\beta_2}\widetilde{\theta}^j(0)^\top \widetilde{\theta}^j(0) \\
&\quad - \frac{(1-\gamma_2)}{2\beta_2}\widetilde{\theta}^{j-1}(T)^\top \widetilde{\theta}^{j-1}(T)
\end{aligned}$$

$$
\begin{aligned}
&= \int_0^T \Big[-\frac{\gamma_1}{2\beta_1} \\
&\qquad \times \operatorname{tr}\left\{ \left(\widetilde{W}^j(t) - \widetilde{W}^{j-1}(t)\right)^\top \left(\widetilde{W}^j(t) - \widetilde{W}^{j-1}(t)\right) \right\} \\
&\qquad - \frac{\gamma_2}{2\beta_2} \left(\widetilde{\theta}(t) - \widetilde{\theta}^{j-1}(t)\right)^\top \left(\widetilde{\theta}(t) - \widetilde{\theta}^{j-1}(t)\right) \Big] dt \\
&\quad + \int_0^T \Big[-e_\phi^j(t)^\top \widetilde{W}^j(t)^\top O^{(4)}\left(\widehat{X}^j(t)\right) \\
&\qquad + \left|e_\phi^j(t)\right| \widetilde{\theta}^j(t)^\top Y^j(t) \Big] dt \\
&\le \int_0^T \Big[-e_\phi^j(t)^\top \widetilde{W}^j(t)^\top O^{(4)}\left(\widehat{X}^j(t)\right) \\
&\qquad + \left|e_\phi^j(t)\right| \widetilde{\theta}^j(t)^\top Y^j(t) \Big] dt.
\end{aligned} \tag{38}
$$

Integrating both sides of (29) from 0 to T gives

$$
\begin{aligned}
&\int_0^T \left[-e_\phi^j(t)^\top \widetilde{W}^j(t)^\top O^{(4)}\left(\widehat{X}^j(t)\right) + \left|e_\phi^j(t)\right| \widetilde{\theta}^j(t)^\top Y^j(t) \right] dt \\
&\quad \le -\int_0^T \left(\lambda e_\phi^j(t)^\top e_\phi^j(t) + e_\phi^j(t)^\top e_\phi^j(t) Y^j(t)^\top Y^j(t) \right) dt \\
&\qquad - \frac{1}{2} e_\phi^j(T)^\top e_\phi^j(T),
\end{aligned} \tag{39}
$$

where we use the property of $(1/2)e_\phi^j(0)^\top e_\phi^j(0) = 0$. Substituting (39) into (38), it yields

$$
\begin{aligned}
&V^j(T) - V^{j-1}(T) \\
&\quad \le -\int_0^T \left(\lambda e_\phi^j(t)^\top e_\phi^j(t) + e_\phi^j(t)^\top e_\phi^j(t) Y^j(t)^\top Y^j(t) \right) dt \\
&\qquad - \frac{1}{2} e_\phi^j(T)^\top e_\phi^j(T).
\end{aligned} \tag{40}
$$

Since $V^1(T)$ is bounded by Lemma 1 and $V^j(T)$ is positive and monotonically decreasing, we conclude by the result of (40) that $V^j(T)$ is bounded for all $j \ge 1$ and will converge as j approaches infinity to some limit value $V(T)$ which is independent of j. The boundedness of $V^j(T)$ also ensures the boundedness of $\widetilde{W}^j(T)$ and $\widetilde{\theta}^j(T)$ for all $j \ge 1$. On the other hand, (40) also implies

$$
\begin{aligned}
\int_0^T e_\phi^j(t)^\top e_\phi^j(t) Y^j(t)^\top Y^j(t)\, dt &\le V^{j-1}(T) - V^j(T) \\
&\le V^1(T),
\end{aligned} \tag{41}
$$

$$
\int_0^T \lambda e_\phi^j(t)^\top e_\phi^j(t)\, dt \le V^{j-1}(T) - V^j(T) \le V^1(T), \tag{42}
$$

$$
\frac{1}{2} e_\phi^j(T)^\top e_\phi^j(T) \le V^{j-1}(T) - V^j(T) \le V^1(T). \tag{43}
$$

It follows that $e_\phi^j(T)^\top e_\phi^j(T)$ are bounded for all $j \ge 1$. Furthermore, $\lim_{j\to\infty} \int_0^T e_\phi^j(t)^\top e_\phi^j(t) dt = 0$ and $\lim_{j\to\infty} e_\phi^j(T)^\top e_\phi^j(T) = 0$. □

Using the boundedness of $\widetilde{W}^j(T)$ and $\widetilde{\theta}^j(T)$ (or equivalently the boundedness of $\widetilde{W}^j(0)$ and $\widetilde{\theta}^j(0)$ for all $j \ge 1$ as shown in Lemma 2, boundedness of all internal signals for all $j \ge 1$ is now established in the following Lemma.

Lemma 3. *Consider the problem setup in Lemma 1. The proposed observer-based adaptive filtered-FNN iterative learning controller ensures that all the internal signals are bounded; that is,* $e_\phi^j(t)$, $e_a^j(t)$, $\widetilde{e}^j$, $y_a^j(t)$, $\widehat{E}^j(t)$, $E^j(t)$, $\widetilde{E}^j$, W^j, $\theta^j(t)$, $u^j(t)$, $\dot{W}^j$, $\dot{\theta}^j \in L_{\infty e}[0,T]$.

Proof. Integrating (34) from 0 to t, we have

$$
\begin{aligned}
V_a^j(t) &\le V_a^j(0) + \int_0^t V_b^{j-1}(t')\, dt' \le V_a^j(0) \\
&\quad + \int_0^T V_b^{j-1}(t)\, dt.
\end{aligned} \tag{44}
$$

Since $V^j(T)$, defined in (36), is bounded for all $j \ge 1$ according to Lemma 2, we conclude that $\int_0^T V_b^{j-1}(t)dt$ is bounded for all $j \ge 1$. Furthermore, the initial value $V_a^j(0)$ is also bounded for all $j \ge 1$ due to Lemma 2. This readily implies from (44) that $V_a^j(t)$ and hence, $e_\phi^j(t)$, $\widetilde{W}^j(t)$, $\widetilde{\theta}^j(t) \in L_{\infty e}[0,T]$. Using the same argument given in Lemma 1, it can be easily shown that $e_a^j(t) \in L_{\infty e}[0,T]$ for all $j \ge 1$.

However, the boundedness of $V_a^j(t)$, $e_\phi^j(t)$, $\widetilde{W}^j(t)$, $\widetilde{\theta}^j(t)$, $e_a^j(t)$, and $\dot{W}^j(t)$ cannot guarantee the boundedness of $Y^j(t)$ (or equivalently $m^j(t)$) and input $u^j(t)$. In order to show the boundedness of $Y^j(t)$ for all $t \in [0,T]$, we first note that $\int_0^T e_\phi^j(t)^\top e_\phi^j(t) Y^j(t)^\top Y^j(t) dt \in L_{\infty e}[0,T]$ are established in (41). In addition, $\int_0^t e_\phi^j(t')^\top e_\phi^j(t') Y^j(t')^\top Y^j(t') dt' \in L_{\infty e}[0,T]$ since $\int_0^t e_\phi^j(t')^\top e_\phi^j(t') Y^j(t')^\top Y^j(t') dt' \le \int_0^T e_\phi^j(t)^\top e_\phi^j(t) Y^j(t)^\top Y^j(t) dt$. But the boundedness of $e_\phi^j(t)$ and $\int_0^t e_\phi^j(t')^\top e_\phi^j(t') Y^j(t')^\top Y^j(t') dt'$ only ensures that $Y^j(t)$ is bounded everywhere except on a set of measure zero. Now we adopt some techniques given in chapter 2 of [26]. Firstly, rewrite $u^j(t)$ in (27) as follows:

$$
\begin{aligned}
u^j(t) &= \frac{L(s)}{F(\tau s)} \left[v^j(t) \right] \\
&= \frac{L(s)}{F(\tau s)} \Big[W^j(t)^\top O^{(4)}\left(\widehat{X}^j(t)\right) \\
&\qquad + \operatorname{sat}\left(\frac{e_a^j(t)}{\phi^j(t)} \right) \theta^j(t)^\top Y^j(t) \\
&\qquad + e_\phi^j(t) Y^j(t)^\top Y^j(t) \Big]
\end{aligned}
$$

$$= \frac{L(s)}{F(\tau s)}\left[W^j(t)^\top O^{(4)}\left(\widehat{X}^j(t)\right) + \operatorname{sat}\left(\frac{e_a^j(t)}{\phi^j(t)}\right)\theta^j(t)^\top Y^j(t)\right] + \frac{sL(s)}{F(\tau s)}\left[\int_0^t e_\phi^j(t')^\top Y^j(t')^\top Y^j(t')\,dt'\right]. \tag{45}$$

Since $W^j(t), O^{(4)}(\widehat{X}^j(t)), \theta^j(t)$, and $\int_0^t e_\phi^j(t')^\top Y^j(t')^\top Y^j(t')dt'$ are bounded for $t \in [0,T]$, and $L(s)/F(\tau s)$, $sL(s)/F(\tau s)$ are proper or strictly proper stable transfer functions, (45) implies that $u^j(t)$ will satisfy

$$\left|u^j(t)\right| \le k_1\left\|\left(Y^j\right)_t\right\|_\infty + k_1 \le k_2\left\|\left(m^j\right)_t\right\|_\infty + k_2 \tag{46}$$

for some $k_1, k_2 > 0$ by lemma 2.6 (output is bounded by truncated L_∞ norm of input for a stable linear system) in [26]. Now we construct an extended dynamic equation by using (15) and (25) as follows:

$$\begin{bmatrix}\dot{\widetilde{E}}^j(t)\\ \dot{m}^j(t)\end{bmatrix} = \begin{bmatrix}A_o & 0\\ 0 & -\delta_1\end{bmatrix}\begin{bmatrix}\widetilde{E}^j(t)\\ m^j(t)\end{bmatrix} + \begin{bmatrix}B\\ 0\end{bmatrix}\left(-b\left(X^j(t)\right)u^j(t) + W^{*\top}O^{(3)}\left(\widehat{X}^j(t)\right) + \delta^j(t)\right) + \begin{bmatrix}0\\ \delta_2\left(1+\left|u^j(t)\right|\right)\end{bmatrix}. \tag{47}$$

Let $X_a^j(t)$ be the state vector of the extended dynamic equation (47). Taking norms on (47) will yield

$$\begin{aligned}\left\|\dot{X}_a^j(t)\right\| &\le k_3\left\|X_a^j(t)\right\| + k_3\left|u^j(t)\right| + k_3\\ &\le k_4\left\|\left(X_a^j\right)_t\right\|_\infty + k_4\end{aligned} \tag{48}$$

for some k_3, $k_4 > 0$. This implies that $X_a^j(t)$ is regular [26] so that $X_a^j(t)$ and hence, $\widetilde{E}^j(t), m^j(t)$ can grow at most exponentially fast and no finite time escape during a finite time interval $[0,T]$. This condition guarantees a certain degree of smoothness of signal $m^j(t)$. Therefore, we show that $Y^j(t)$ is a certain degree of smooth signal. Together with the result of $\int_0^t e_\phi^j(t')Y^j(t')^\top Y^j(t')dt' \in L_{\infty e}[0,T]$, we can now conclude that $Y^j(t) \in L_{\infty e}[0,T]$.

Due to $W^j(t), O^{(4)}(\widehat{X}^j(t)), \theta^j(t), Y^j(t), e_\phi^j(t) \in L_{\infty e}[0,T]$, this also implies that $v^j(t) \in L_{\infty e}[0,T]$ (by (28)). Since $e_\phi^j(t)$, $W^j(t)$, $\theta^j(t)$, $O^{(4)}(\widehat{X}^j(t))$, $Y^j(t) \in L_{\infty e}[0,T]$, we conclude that $\dot{W}^j(t)$ (by (30)), $\dot{\theta}^j(t)$ (by (31)) $\in L_{\infty e}[0,T]$. Since $v^j(t) \in L_{\infty e}[0,T]$ and $L(s)/F(\tau s)$ is a proper stable transfer function, this implies that $u^j(t) \in L_{\infty e}[0,T]$ (by (27)). Since $v^j(t), u^j(t) \in L_{\infty e}[0,T]$, $1/L(s)$ and $1/\ell(s)$ are strictly proper stable transfer functions, this implies that $y_a^j(t) \in L_{\infty e}[0,T]$ (by (18)). As noted above, $e_a^j(t), y_a^j(t) \in L_{\infty e}[0,T]$, we have $\widetilde{e}^j(t) \in L_{\infty e}[0,T]$ (by (19)). Moreover, because A_c is a Hurwitz matrix and $\widetilde{e}^j \in L_{\infty e}[0,T]$, this implies that $\widetilde{E}^j(t) \in L_{\infty e}[0,T]$ (by (12)). Finally, since A_o are Hurwitz matrices and $-b(X^j(t))u^j(t) + W^{*\top}O^{(3)}(\widehat{X}^j(t)) + \delta^j(t) \in L_{\infty e}[0,T]$, we have $\widetilde{E}^j \in L_{\infty e}[0,T]$ (by (14)). Since $\widehat{E}^j(t), \widetilde{E}^j(t) \in L_{\infty e}[0,T]$, it implies $E^j(t) \in L_{\infty e}[0,T]$ (by (13)). This completes the proof. □

Based on Lemmas 1, 2, and 3, we now state the main result in the following theorem.

Theorem 4. *Consider the system setup in Lemma 1. The proposed observer-based adaptive filtered-FNN iterative learning controller guarantees the tracking performance and system stability as follows:*

(T1) $\lim_{j\to\infty} e_\phi^j(t)^\top e_\phi^j(t) = 0$, *for all* $t \in [0,T]$.

(T2) $\lim_{j\to\infty}|e_a^j(t)| \le e^{-\lambda t}\varepsilon^\infty$, *for all* $t \in [0,T]$.

(T3) $\lim_{j\to\infty}|\widetilde{e}^j(t)| \le e^{-\lambda t}\varepsilon^\infty + \tau k_6 5$, *for all* $t \in [0,T]$ *and for a constant* $k_5 > 0$.

(T4) *Let* δ *and* k_6 *be the positive constants such that the transition matrix* $\Phi(t)$ *of* A_c *satisfies* $|\Phi(t)| \le k_6 e^{-\delta t}$. *Then there exists a positive constant* k_7 *such that* $\lim_{j\to\infty}|\widetilde{e}^j(t)| = |\widetilde{e}^\infty(t)| \le k_7(\varepsilon^\infty((e^{-\lambda t} - e^{-\delta t})/(\delta - \lambda)) + \tau k_5((1-e^{-\delta t})/\delta))$, *for all* $t \in [0,T]$.

(T5) $\lim_{j\to\infty}|e^j(t)| = |e^\infty(t)| \le e^{-\lambda t}\varepsilon^\infty + \tau k_5 + k_7(\varepsilon^\infty((e^{-\lambda t} - e^{-\delta t})/(\delta - \lambda)) + \tau k_5((1-e^{-\delta t})/\delta))$, *for all* $t \in [0,T]$.

Proof. (T1) Since $e_\phi^j(t)^\top e_\phi^j(t) \in L_{\infty e}[0,T]$ and (d/dt) $e_\phi^j(t)^\top e_\phi^j(t) = 2e_\phi^j(t)^\top(\dot{e}_a^j(t) - \operatorname{sgn}(e_\phi^j(t))\dot{\phi}^j(t)) \in L_{\infty e}[0,T]$ for all $j \ge 1$. These facts imply that $e_\phi^j(t)^\top e_\phi^j(t)$ is uniformly continuous over $[0,T]$ for all $j \ge 1$. On the other hand, $e_\phi^j(t)^\top e_\phi^j(t)$ satisfies $\lim_{j\to\infty}\int_0^T e_\phi^j(t)^\top e_\phi^j(t)dt = 0$ due to the result of Lemma 2. We can now conclude, by using similar argument for Barbalat's lemma (e.g., Lemma 3.2.6 in [28]), that $\lim_{j\to\infty} e_\phi^j(t)^\top e_\phi^j(t) = 0$ for all $t \in [0,T]$.

(T2) According to the definition of $e_\phi^j(t)$ in (22), we can derive the bound of

$$\lim_{j\to\infty}\left|e_a^j(t)\right| = \left|e_a^\infty(t)\right| \le \phi^\infty(t) = e^{-\lambda t}\varepsilon^\infty \tag{49}$$

for all $t \in [0,T]$.

(T3) Substituting (27) into (18), we can find that $e_a^j(t)$ actually satisfies

$$\begin{aligned} e_a^j(t) &= \tilde{e}^j(t) - y_a^j(t) \\ &= \tilde{e}^j(t) - \frac{1}{\ell(s)}\left[v^j(t) - \frac{1}{L(s)}\left[u^j(t)\right]\right] \\ &= \tilde{e}^j(t) - \frac{1}{\ell(s)}\left[v^j(t) - \frac{1}{F(\tau s)}\left[v^j(t)\right]\right] \\ &= \tilde{e}^j(t) - \frac{1}{\ell(s)}\left(1 - \frac{1}{F(\tau s)}\right)\left[v^j(t)\right] \\ &\triangleq \tilde{e}^j(t) - R^j(t). \end{aligned} \tag{50}$$

Since $v^j(t)$ is bounded and the H_∞ norm of $\|(1/s)(1-1/F(\tau s))\|_\infty = 2\tau$ and $\|s/\ell(s)\|_\infty = \max\{1, 1/\lambda\}$, we can conclude that

$$\left|R^j(t)\right| \le \left\|\frac{1}{s}\left(1 - \frac{1}{F(\tau s)}\right)\right\|_\infty \left\|\frac{s}{\ell(s)}\right\|_\infty \left\|\left(v^j(t)\right)_t\right\|_\infty \le \tau k_5 \tag{51}$$

for some $k_5 > 0$. Taking norms on (50), we find that

$$\left|\tilde{e}^j(t)\right| \le \left|e_a^j(t)\right| + \left|R^j(t)\right| \le \left|e_a^j(t)\right| + \tau k_5. \tag{52}$$

As iteration goes to infinity,

$$\lim_{j\to\infty}\left|\tilde{e}^j(t)\right| \le e^{-\lambda t}\varepsilon^\infty + \tau k_5. \tag{53}$$

(T4) Consider the following tracking error estimation dynamics as $j \to \infty$,

$$\begin{aligned} \dot{\hat{E}}^\infty(t) &= A_c\hat{E}^\infty(t) + K_o\tilde{e}^\infty(t), \quad \hat{E}^\infty(0) = 0, \\ \hat{e}^\infty(t) &= C^\top\hat{E}^\infty(t). \end{aligned} \tag{54}$$

The solution of (54) in time domain is given by

$$\hat{e}^\infty(t) = \int_0^t C^\top\Phi\left(t-t'\right)K_o\tilde{e}^\infty\left(t'\right)dt'. \tag{55}$$

Taking norms on (55), it yields

$$\begin{aligned} &\left|\hat{e}^\infty(t)\right| \\ &\le k_7\int_0^t e^{-\delta(t-t')}\left|\tilde{e}^\infty\left(t'\right)\right|dt' \\ &\le k_7\int_0^t e^{-\delta(t-t')}\left(e^{-\lambda t'}\varepsilon^\infty + \tau k_5\right)dt' \\ &\le k_7\int_0^t e^{-\delta(t-t')}e^{-\lambda t'}\varepsilon^\infty dt' + k_7\int_0^t e^{-\delta(t-t')}\tau k_5 dt' \\ &\le k_7\left(\varepsilon^\infty\frac{e^{-\lambda t} - e^{-\delta t}}{\delta - \lambda} + \tau k_6\frac{1-e^{-\delta t}}{\delta}\right) \end{aligned} \tag{56}$$

for some positive constant $k_7 = k_6|C \parallel K_o|$.

(T5) Finally, we investigate the tracking performance in the final iteration when (T1), (T2), (T3), and (T4) of this theorem are achieved. Since $\tilde{e}^j(t) = e^j(t) - \hat{e}^j(t)$, we have

$$\left|e^j(t)\right| \le \left|\tilde{e}^j(t)\right| + \left|\hat{e}^j(t)\right|. \tag{57}$$

As iteration goes to infinity,

$$\begin{aligned} &\lim_{j\to\infty}\left|e^j(t)\right| \\ &\le \lim_{j\to\infty}\left|\tilde{e}^j(t)\right| + \lim_{j\to\infty}\left|\hat{e}^j(t)\right| \\ &\le e^{-\lambda t}\varepsilon^\infty + \tau k_5 + k_7\left(\varepsilon^\infty\frac{e^{-\lambda t} - e^{-\delta t}}{\delta - \lambda} + \tau k_5\frac{1-e^{-\delta t}}{\delta}\right) \end{aligned} \tag{58}$$

for $t \in [0, T]$. This completes the proof. □

Remark 5. In Theorem 4, we show that the output learning error $e^j(t)$ will converge to a residual set as the number of iterations approaches infinity. In general, the size of the residual set can be tuned by two design parameters. The first one λ is the decay rate of the boundary layer $\phi^j(t) = \varepsilon^j e^{-\lambda t}$ which will be chosen as large as possible. The second one τ is the filter parameter of the averaging filter $1/F(\tau s) = 1/(\tau s + 1)^2$ which will be chosen as small as possible. To guarantee a satisfied learning performance, we will usually choose a larger λ and smaller τ. Furthermore, if there is no initial output error, that is, $\varepsilon^j = 0$, the output learning error will converge to a residual set whose level of magnitude depends on τ only.

5. Simulation Example

In this section, a computer simulation is conducted to demonstrate the learning effect of the proposed observer based adaptive filtered-FNN iterative learning controller. Here we consider a two-link planar robotic system [29] with the dynamic equation of

$$\begin{aligned} &\begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix}\begin{bmatrix} \ddot{q}_1^j(t) \\ \ddot{q}_2^j(t) \end{bmatrix} \\ &\quad + \begin{bmatrix} -h\dot{q}_2^j(t) & -h\left(\dot{q}_1^j(t) + \dot{q}_2^j(t)\right) \\ h\dot{q}_1^j(t) & 0 \end{bmatrix}\begin{bmatrix} \dot{q}_1^j(t) \\ \dot{q}_2^j(t) \end{bmatrix} \end{aligned} \tag{59}$$

where $D_{11} = m_1 l_{c1}^2 + m_2(l_1^2 + l_{c2}^2 + 2l_1 l_{c2}\cos(q_2^j(t))) + I_1 + I_2$, $D_{12} = D_{21} = m_2 l_1 l_{c2}\cos(q_2^j(t)) + m_2 l_{c2}^2 + I_2$, $D_{22} = m_2 l_{c2}^2 + I_2$, and $h = m_2 l_1 l_{c2}\sin(q_2^j(t))$. Here m_i, I_i, l_i, and l_{c_i} represent mass, inertia, length of link i, and the distance from the previous joint to the center of mass of link i, respectively.

In this simulation, we set $m_1 = 10\,\text{kg}$, $m_2 = 5\,\text{kg}$, $l_1 = 1\,\text{m}$, $l_2 = 0.5\,\text{m}$, $l_{c1} = 0.5\,\text{m}$, $l_{c2} = 0.25\,\text{m}$, $I_1 = 0.83\,\text{kg-m}^2$, and $I_2 = 0.3\,\text{kg-m}^2$. The control objective is to let $q^j(t) = [q_1^j(t), q_2^j(t)]^\top$ track the desired trajectory $q_d(t) = [q_{d_1}(t), q_{d_2}(t)]^\top = [\sin(t), \cos(t)]^\top$ as close as possible over a finite time interval $[0, 15]$ when only the angular position $q^j(t)$ is measurable. The design steps are summarized as follows.

(S1) Specify the observer and feedback gain vectors $K_o = [k_1^o I_{2\times 2}, k_2^o I_{2\times 2}]^\top = [16I_{2\times 2}, 9I_{2\times 2}]^\top \in \mathcal{R}^{4\times 2}$, $K_c = [k_2^c I_{2\times 2}, k_1^c I_{2\times 2}]^\top = [4I_{2\times 2}, 4I_{2\times 2}]^\top \in \mathcal{R}^{4\times 2}$, respectively, such that the matrices $A_c = A - BK_c^\top$ and $A_o = A_c - K_o C^\top$ are Hurwitz.

(S2) Design the tracking error observer as in (12) to obtain $\widehat{e}^j(t)$, $\widetilde{e}^j(t) = e^j(t) - \widehat{e}^j(t)$ and the state estimation vector $\widehat{X}^j(t) = X_d(t) - \widehat{E}^j(t)$.

(S3) Select $H(s) = (1/\ell(s)L(s))I_{2\times 2}$ with $\ell(s) = s + \lambda = s + 10$ and $L(s) = s + 10$. Then, define an augmented signal with filtered version as $y_a^j(t) = (1/\ell(s))[v^j(t) - (1/L(s))[u^j(t)]]$, $y_a^j(0) = 0$, an auxiliary error signal as $e_a^j(t) = \widetilde{e}^j(t) - y_a^j(t)$, and a dead-zone signal $e_\phi^j(t) = e_a^j(t) - \phi^j(t)\,\mathrm{sat}(e_a^j(t)/\phi^j(t))$ with $\phi^j(t) = \varepsilon^j e^{-\lambda t} = \varepsilon^j e^{-10t}$. The normalization signal is given as $m^j(t) = (\delta_2/(s+\delta_1))[1 + |u^j(t)|]$, $m^j(0) > \delta_2/\delta_1$ with $\delta_1 = \delta_2 = 0.01$.

(S4) Construct the membership functions for $\widehat{X}^j(t)$. Then, solve the filtered basis function vector $O^{(4)}(\widehat{X}^j(t))$. The filtered-FNN with input $\widehat{X}^j(t)$ is given as $W^j(t)^\top O^{(4)}(\widehat{X}^j(t))$. Since the working domain of the desired trajectory $X_d(t) = [q_{d_1}(t), q_{d_2}(t), \dot{q}_{d_1}(t), \dot{q}_{d_2}(t)]^\top = [\sin(t), \cos(t), \cos(t), -\sin(t)]^\top$ is within the interval $[-1, 1]$, we choose the centers as $m = [m_1, m_2, m_3, m_4]$ with $m_i = [m_{i1}, m_{i2}, m_{i3}, m_{i4}, m_{i5}] = [-1, -0.5, 0, 0.5, 1]$, $i = 1, 2, 3, 4$ and variances as $\sigma = [\sigma_1, \sigma_2, \sigma_3, \sigma_4]$ with $\sigma_i = [\sigma_{i1}, \sigma_{i2}, \sigma_{i3}, \sigma_{i4}, \sigma_{i5}] = [0.25, 0.25, 0.25, 0.25, 0.25]$, $i = 1, 2, 3, 4$ to cover this interval, respectively. In addition, we set the control parameter $\theta^0(t) = \theta^0 = 0.1$ at the first iteration for all $t \in [0, 15]$. It is noted that the initial values of the consequent parameters $W^0(t)$ can be roughly estimated if the nonlinear function $h(X^j(t))$ of the robotic system is partially known. However, we often arbitrarily choose this initial parameters.

(S5) Design the observer-based adaptive filtered-FNN iterative learning controller as $u^j(t) = (\mathrm{L}(s)/F(\tau s))[v^j(t)]$ where $v^j(t) = W^j(t)^\top O^{(4)}(\widehat{X}^j(t)) + \mathrm{sat}(e_a^j(t)/\phi^j(t))\theta^j(t)^\top Y^j(t) + e_\phi^j(t) Y^j(t)^\top Y^j(t)$. The averaging filter is designed with $\tau = 0.0001$.

(S6) Finally, the adaptation algorithms (30) and (31) are adopted to update the filtered-FNN parameters and control parameters with $\gamma_1 = \gamma_2 = 0.5$, $\beta_1 = \beta_2 = 500$.

In order to show the robustness to the varying initial output tracking error, we first assume the initial joint position vector of the two-link planar robotic system taking the following arbitrary values for the first five iterations: $q^j(0) = [q_1^j(0), q_2^j(0)]^\top = [0.05, 0.3]^\top$, $[0.25, 0.8]^\top$, $[0.14, 0.6]^\top$, $[0.09, 0.4]^\top$, $[-0.1, 0.5]^\top$. At the beginning of each iteration, the initial value of the boundary layer $\phi^j(t)$ is then chosen according to $\phi^j(0) = \varepsilon^j = |[e_{a,1}^j(0), e_{a,2}^j(0)]^\top| = |[\widetilde{e}_1^j(0), \widetilde{e}_2^j(0)]^\top| = |[e_1^j(0), e_2^j(0)]^\top| = |[\sin(0) - q_1^j(0), \cos(0) - q_2^j(0)]^\top|$. For example, $\phi^1(0) = 0.05$, $\phi^2(0) = 0.25$, $\phi^3(0) = 0.14$, $\phi^4(0) = 0.09$, and $\phi^5(0) = 0.1$. To study the effect of learning performances, $\sup_{t\in[0,15]}|e_{\phi,1}^j(t)|$ and $\sup_{t\in[0,15]}|e_{\phi,2}^j(t)|$ versus iteration j are shown in Figures 1(a) and 1(b), respectively. It is clear that the asymptotic convergence proved in (T1) of Theorem 4 is achieved. Since the learning process is almost completed at the 5th iteration, we demonstrate the auxiliary errors $e_{a,1}^5(t)$ and $e_{a,2}^5(t)$ in Figures 2(a) and 2(b), respectively. The trajectories of $e_{a,1}^5(t)$ and $e_{a,2}^5(t)$ satisfy $-0.5e^{-10t} \le e_{a,1}^5(t) \le 0.5e^{-10t}$ and $-0.5e^{-10t} \le e_{a,2}^5(t) \le 0.5e^{-10t}$, respectively. This clearly proves (T2) of Theorem 4. The output tracking errors $e_1^5(t)$ and $e_2^5(t)$ are shown in Figures 3(a) and 3(b) with satisfied performance even there exists varying initial output errors. The nice tracking performances at the 5th iteration between joint position vector $q^j(t) = [q_1^j(t), q_2^j(t)]^\top$ and desired joint position vector $q_d(t) = [q_{d,1}(t), q_{d,2}(t)]^\top$ are presented in Figures 4(a) and 4(b), respectively. Finally, the bounded learned control forces $u_1^5(t)$ and $u_2^5(t)$ are plotted in Figures 5(a) and 5(b), respectively.

6. Conclusion

An observer-based adaptive filtered-FNN iterative learning controller for repeated tracking control of uncertain robotic systems is proposed in this paper. A tracking error observer is designed to estimate the unknown joint variables since only joint positions are assumed to be measurable. An error observation dynamic based on the tracking error observer is derived for the design of the iterative learning controller. The main control force is designed by using a technique of averaging filter and some auxiliary signals. In this main control force, a fuzzy neural learning component based on a filtered-FNN is used to approximate the unknown nonlinear function, a robust learning component is designed to compensate for the other uncertainties, and a stabilization term is applied to guarantee the boundedness of internal signals. The adaptive laws combining time domain and iteration domain adaptation for the network parameters and control parameters are proposed to ensure the stability and convergence of the learning system. A Lyapunov-like analysis has been developed to solve both boundedness of internal signals inside the closed-loop system and asymptotic convergence of learning error. It is shown that the output tracking error asymptotically converges to a tunable residual set whose level of magnitude can be tuned by design parameters as the iteration number increases.

Appendix

Filtered Fuzzy Neural Network

The structure of the filtered-FNN is shown in Figure 6. The only difference between the filtered-FNN and traditional FNN is that there is a filter layer between the output layer

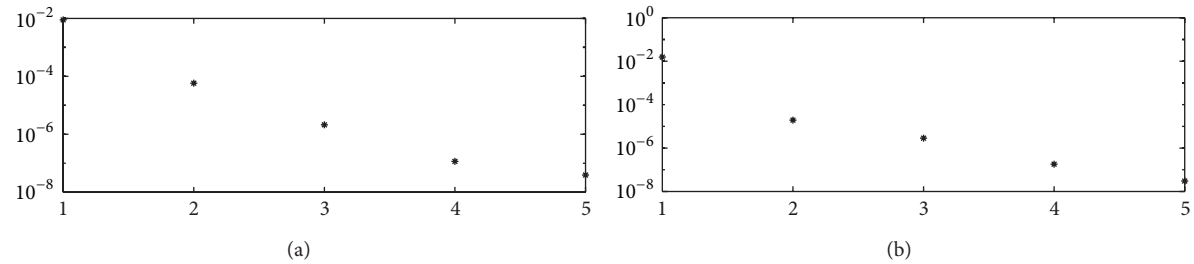


Figure 1: (a) $\sup_{t\in[0,15]}|e^{j}_{\phi,1}(t)|$ versus iteration j; (b) $\sup_{t\in[0,15]}|e^{j}_{\phi,2}(t)|$ versus iteration j.

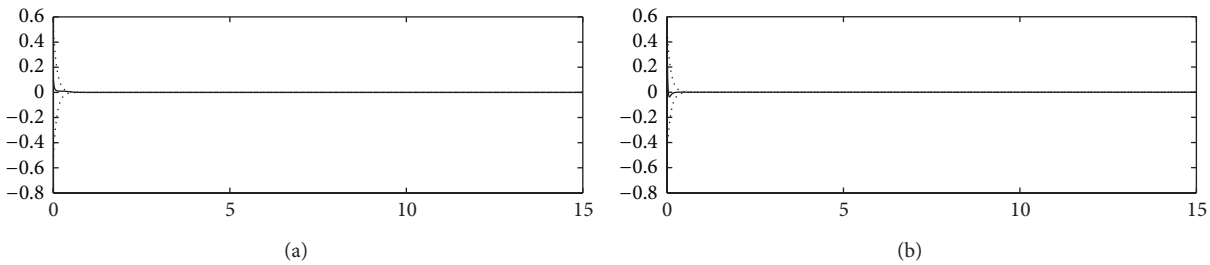


Figure 2: (a) $e^{5}_{a,1}(t)$ (solid line) and $\pm\phi^{5}(t)$ (dotted lines) versus time t; (b) $e^{5}_{a,2}(t)$ (solid line) and $\pm\phi^{5}(t)$ (dotted lines) versus time t.

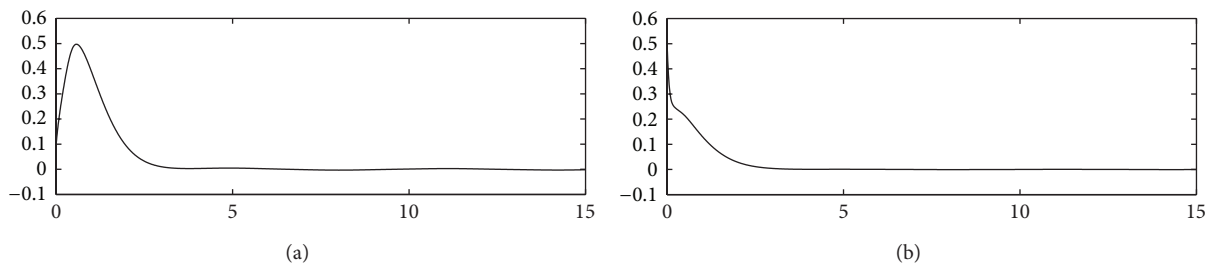


Figure 3: (a) $e^{5}_{1}(t)$ versus time t; (b) $e^{5}_{2}(t)$ versus time t.

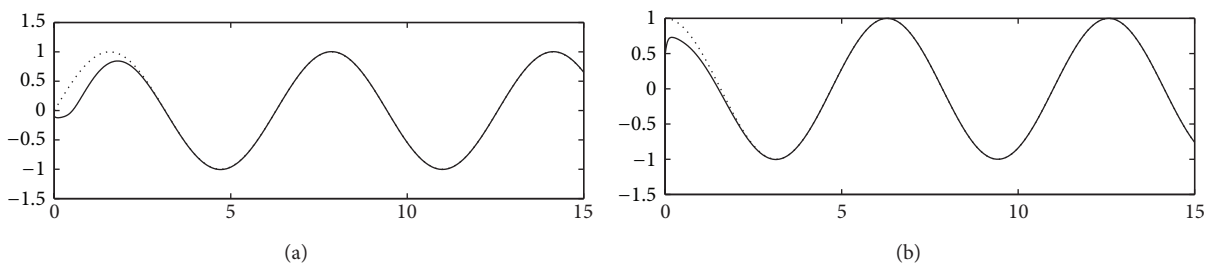


Figure 4: (a) $q^{5}_{1}(t)$ (solid line) and $q_{d,1}(t)$ (dotted line) versus time t; (b) $q^{5}_{2}(t)$ (solid line) and $q_{d,2}(t)$ (dotted line) versus time t.

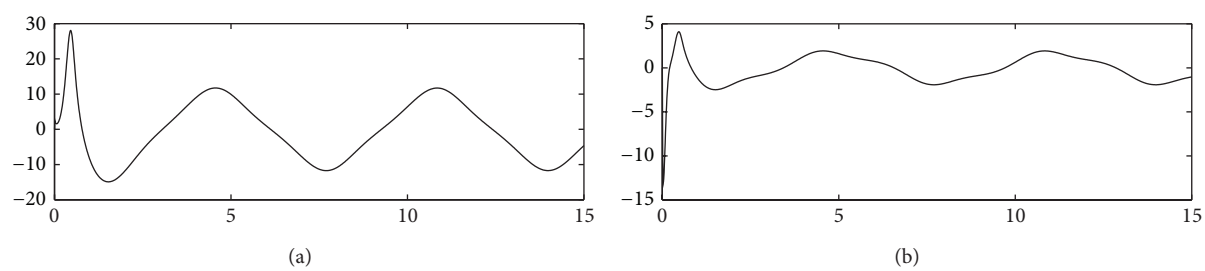


Figure 5: (a) $u^{5}_{1}(t)$ versus time t; (b) $u^{5}_{2}(t)$ versus time t.

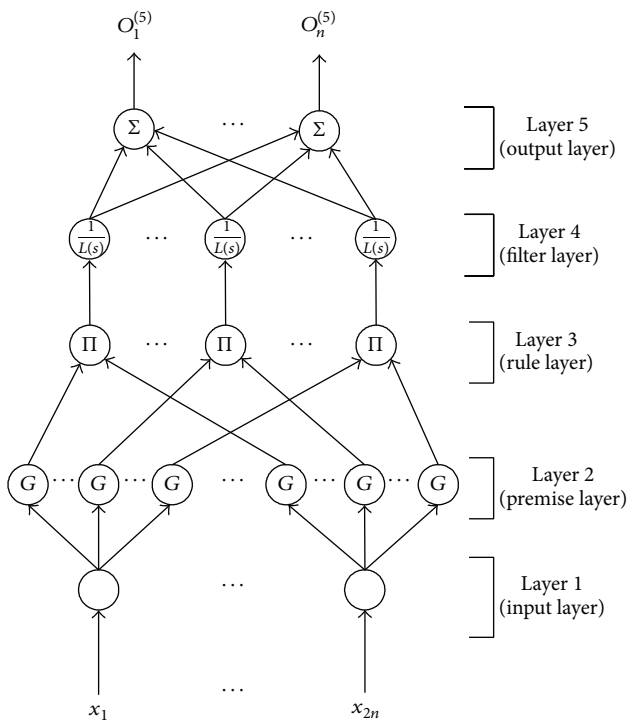


Figure 6: Configuration of the filtered fuzzy neural network (filtered-FNN).

(Layer 5) and rule layer (Layer 3). In other words, the basis function in filtered-FNN is defined as the output of the basis function vector in traditional FNN through a stable filter. The layered operation of the filtered-FNN is briefly described. It is noted that dependency on the iteration index j and time index t will be omitted unless emphasis on the temporal or iterative relationship is required. The layered operation of the filtered-FNN is briefly described below.

Layer 1 (input layer). Each node in this layer represents an input linguistic variable, which only transmits input value to the next layer directly. For the ith input node, $i = 1, \ldots, 2n$,

$$\mathrm{net}_i^{(1)} = x_i; \quad O_i^{(1)} = f_i^{(1)}\left(\mathrm{net}_i^{(1)}\right) = \mathrm{net}_i^{(1)}, \tag{A.1}$$

where x_i represents the external input signal to the ith input node of Layer 1.

Layer 2 (premise layer). Each node in this layer represents a membership function. In this layer, the Gaussian membership function is adopted as a membership function such that the ℓth term node of the ith input linguistic variable, $i = 1, \ldots, 2n$, $\ell = 1, \ldots, M$, will be represented as

$$\begin{aligned} \mathrm{net}_{i\ell}^{(2)} &= -\left(\frac{x_i^{(2)} - m_{i\ell}}{\sigma_{i\ell}}\right)^2; \\ O_{i\ell}^{(2)} &= f_{i\ell}^{(2)}\left(\mathrm{net}_{i\ell}^{(2)}\right) = \exp\left(\mathrm{net}_{i\ell}^{(2)}\right), \end{aligned} \tag{A.2}$$

where $x_i^{(2)} = O_i^{(1)}$, $m_{i\ell}$, and $\sigma_{i\ell}$ are the mean and variance of the Gaussian membership function.

Layer 3 (rule layer). Each node in this layer represents the firing strength of a fuzzy rule. The fuzzy reasoning is performed by product of the input signals of the ℓth rule node, which is represented as

$$\mathrm{net}_\ell^{(3)} = \prod_{i=1}^{2n} x_{i\ell}^{(3)}; \qquad O_\ell^{(3)} = f_\ell^{(3)}\left(\mathrm{net}_\ell^{(3)}\right) = \mathrm{net}_\ell^{(3)}, \tag{A.3}$$

where $x_{i\ell}^{(3)} = O_{i\ell}^{(2)}$ is the $i\ell$th input to the ℓth rule node of Layer 3.

Layer 4 (filter layer). Each node in this layer represents the filtered firing strength. The filtered firing strength is the output of the firing strength through a stable filter $1/L(s)$. For the ℓth filter node, $\ell = 1, \ldots, M$,

$$\mathrm{net}_\ell^{(4)} = \frac{1}{L(s)}\left[x_\ell^{(4)}\right]; \qquad O_\ell^{(4)} = f_\ell^{(4)}\left(\mathrm{net}_\ell^{(4)}\right) = \mathrm{net}_\ell^{(4)}, \tag{A.4}$$

where $x_\ell^{(4)} = O_\ell^{(3)}$ represents the ℓth input signal to the ℓth node of Layer 4.

Layer 5 (output layer). This layer performs the singleton defuzzification. The node in this layer represents an output node, which is the consequence with respect to the oth output variable from each rule. For the oth output node, $o = 1, \ldots, n$,

$$\mathrm{net}_o^{(5)} = \sum_{\ell=1}^{M} w_{\ell o} x_\ell^{(5)}; \qquad O_o^{(5)} = f_o^{(5)}\left(\mathrm{net}_o^{(5)}\right) = \mathrm{net}_o^{(5)}, \tag{A.5}$$

where $x_\ell^{(5)} = O_\ell^{(4)}$ is the ℓth input to the output node and the link weight $w_{\ell o}$ is the action strength of the oth output associated with the ℓth rule node.

The overall representation from external input x_i in Layer 1 to the oth output $O_o^{(5)}$ in Layer 5 is then given by

$$O_o^{(5)} = \sum_{\ell=1}^{M} w_{\ell o} \cdot \frac{1}{L(s)}\left[\prod_{i=1}^{2n} \exp\left\{-\left(\frac{x_i - m_{i\ell}}{\sigma_{i\ell}}\right)^2\right\}\right]. \tag{A.6}$$

Let the output and input of the filtered-FNN be $O^{(5)} = [O_1^{(5)}, \ldots, O_n^{(5)}]^\top \in \mathscr{R}^{n\times 1}$ and $X = [x_1, \ldots, x_{2n}]^\top \in \mathscr{R}^{2n\times 1}$, respectively. From the representation (A.6), a multiinput multi-output filtered-FNN can be further described in a matrix form as follows:

$$O^{(5)}(X) = O^{(5)}(X, W) = W^\top \cdot O^{(4)}(X), \tag{A.7}$$

where $W \in \mathscr{R}^{M\times n}$ is the weight matrix of output layer and $O^{(4)}(X) \in \mathscr{R}^{M\times 1}$ is the filtered basis function vector with M being the number of rule nodes.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work is supported by the National Science Council under Grants NSC102-2221-E-211-003 and NSC102-2221-E-211-011.

References

[1] K. L. Moore and X. U. Jian-Xin, "Special issue on iterative learning control," *International Journal of Control*, vol. 73, no. 10, pp. 819–823, 2000.

[2] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control: a learning-based method for high-performance tracking control," *IEEE Control Systems Magazine*, vol. 26, no. 3, pp. 96–114, 2006.

[3] H.-S. Ahn, Y. Q. Chen, and K. L. Moore, "Iterative learning control: brief survey and categorization," *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 37, no. 6, pp. 1099–1121, 2007.

[4] Y. Wang, F. Gao, and F. J. Doyle III, "Survey on iterative learning control, repetitive control, and run-to-run control," *Journal of Process Control*, vol. 19, no. 10, pp. 1589–1600, 2009.

[5] J.-X. Xu, "A survey on iterative learning control for nonlinear systems," *International Journal of Control*, vol. 84, no. 7, pp. 1275–1294, 2011.

[6] R. Horowitz, "Learning control of robot manipulators," *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. 115, no. 2 B, pp. 402–411, 1993.

[7] D. Wang, Y. C. Soh, and C. C. Cheah, "Robust motion and force control of constrained manipulators by learning," *Automatica*, vol. 31, no. 2, pp. 257–262, 1995.

[8] C.-J. Chien and J.-S. Liu, "A P-type iterative learning controller for robust output tracking of nonlinear time-varying systems," *International Journal of Control*, vol. 64, no. 2, pp. 319–334, 1996.

[9] F. Bouakrif, "D-type iterative learning control without resetting condition for robot manipulators," *Robotica*, vol. 29, no. 7, pp. 975–980, 2011.

[10] A. Tayebi, "Adaptive iterative learning control for robot manipulators," *Automatica*, vol. 40, no. 7, pp. 1195–1203, 2004.

[11] C.-J. Chien and A. Tayebi, "Further results on adaptive iterative learning control of robot manipulators," *Automatica*, vol. 44, no. 3, pp. 830–837, 2008.

[12] X.-G. Jia and Z.-Y. Yuan, "Adaptive iterative learning control for robot manipulators," in *Proceedings of the IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS '10)*, pp. 139–142, Xiamen, China, October 2010.

[13] J.-M. Wei and Y.-A. Hu, "Adaptive iterative learning control for robot manipulators with initial resetting errors," *Applied Mechanics and Materials*, vol. 130-134, pp. 265–269, 2012.

[14] T. Ngo, Y. Wang, T. L. Mai, J. Ge, M. H. Nguyen, and S. N. Wei, "An adaptive iterative learning control for robot manipulator in task space," *International Journal of Computers, Communications and Control*, vol. 7, no. 3, pp. 510–521, 2012.

[15] A. Tayebi and C.-J. Chien, "A unified adaptive iterative learning control framework for uncertain nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 10, pp. 1907–1913, 2007.

[16] W. Chen, J. M. Li, and J. Li, "Practical adaptive iterative learning control framework based on robust adaptive approach," *Asian Journal of Control*, vol. 13, no. 1, pp. 85–93, 2011.

[17] C.-J. Chien, "A combined adaptive law for fuzzy iterative learning control of nonlinear systems with varying control tasks," *IEEE Transactions on Fuzzy Systems*, vol. 16, no. 1, pp. 40–51, 2008.

[18] Y.-C. Wang and C.-J. Chien, "Decentralized adaptive fuzzy neural iterative learning control for nonaffine nonlinear interconnected systems," *Asian Journal of Control*, vol. 13, no. 1, pp. 94–106, 2011.

[19] Y. C. Wang and C. J. Chien, "Repetitive tracking control of nonlinear systems using reinforcement fuzzy-neural adaptive iterative learning controller," *Applied Mathematics and Information Sciences*, vol. 6, no. 3, pp. 473–481, 2012.

[20] A. Tayebi and J.-X. Xu, "Observer-based iterative learning control for a class of time-varying nonlinear systems," *IEEE Transactions on Circuits and Systems I*, vol. 50, no. 3, pp. 452–455, 2003.

[21] J.-X. Xu and J. Xu, "Observer based learning control for a class of nonlinear systems with time-varying parametric uncertainties," *IEEE Transactions on Automatic Control*, vol. 49, no. 2, pp. 275–281, 2004.

[22] J. Wallén, M. Norrlöf, and S. Gunnarsson, "A framework for analysis of observer-based ILC," *Asian Journal of Control*, vol. 13, no. 1, pp. 3–14, 2011.

[23] Y.-Y. Du, J. S.-H. Tsai, S.-M. Guo, T.-J. Su, and C.-W. Chen, "Observer-based iterative learning control with evolutionary programming algorithm for MIMO nonlinear systems," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 3, pp. 1357–1374, 2011.

[24] W.-S. Chen, R.-H. Li, and J. Li, "Observer-based adaptive iterative learning control for nonlinear systems with time-varying delays," *International Journal of Automation and Computing*, vol. 7, no. 4, pp. 438–446, 2010.

[25] F. Bouakrifa, D. Boukhetalab, and F. Boudjemab, "Velocity observer-based iterative learning control for robot manipulators," *International Journal of Systems Science*, vol. 44, no. 2, pp. 214–222, 2013.

[26] A. M. Annaswamy and K. S. Narendra, *Stable Adaptive Systems*, Prentice-Hall, 1988.

[27] P. A. Ioannou and K. S. Tsakalis, "A robust direct adaptive control," *IEEE Transactions on Automatic Control*, vol. 31, no. 11, pp. 1033–1043, 1986.

[28] P. A. Ioannou and J. Sun, *Robust Adaptive Control*, Prentice Hall, Englewood Cliffs, NJ, USA.

[29] J. J. Slotine and W. Li, *Applied Nonlinear Control*, Prentice-Hall, Englewood Cliffs, NJ, USA.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2014, Article ID 589345, 17 pages
http://dx.doi.org/10.1155/2014/589345

Research Article

Conflict Resolution for Product Performance Requirements Based on Propagation Analysis in the Extension Theory

Yanwei Zhao,[1] Huanhuan Hong,[1] Gaochao Jiang,[1] Weigang Chen,[1] and Hongwei Wang[2]

[1] *Key Laboratory of Special Purpose Equipment and Advanced Manufacturing Technology, Ministry of Education/Zhejiang Province, Hangzhou 310014, China*
[2] *School of Engineering, University of Portsmouth, Portsmouth PO1 3DJ, UK*

Correspondence should be addressed to Yanwei Zhao; zyw@zjut.edu.cn

Received 9 August 2013; Accepted 23 September 2013; Published 29 January 2014

Academic Editor: Sheng-yong Chen

Traditional product data mining methods are mainly focused on the static data. Performance requirements are generally met as possible by finding some cases and changing their structures. However, when one is satisfied with the structures changed, the other effects are not taken into account by analyzing the correlations; that is, design conflicts are not identified and resolved. An approach to resolving the conflict problems is proposed based on propagation analysis in Extension Theory. Firstly, the extension distance is improved to better fit evaluating the similarity among cases, then, a case retrieval method is developed. Secondly, the transformations that can be made on selected cases are formulated by understanding the conflict natures in the different performance requirements, which leads to the extension transformation strategy development for coordinating conflicts using propagation analysis. Thirdly, the effects and levels of propagation are determined by analyzing the performance values before and after the transformations, thus the co-existing conflict coordination strategy of multiple performances is developed. The method has been implemented in a working prototype system for supporting decision-making. And it has been demonstrated the feasible and effective through resolving the conflicts of noise, exhaust, weight and intake pressure for the screw air compressor performance design.

1. Introduction

Product performance not only holds the key to meeting customers' requirements but also is a factor determining enterprises' competitive advantages. Performance Driven Design (PDD) has become an important branch of the modern design theory. Product performance must be taken into account throughout the design process, and PDD becomes increasingly complicated and systematic with the design problems for different performances interacting with, and depending upon, each other. Study on the product performance can further attain its competitive advantages amongst products, which belong to the same function, the same type, or the same grade. A product's overall performance is essentially determined by the degrees of how all the customers' expectations are met as a whole. It is of significant importance to study the correlations among the various factors that influence a product's overall performance, the resolution of overall performance optimization conflict problems caused by meeting different performance requirements. Consequently, the accurate identification and effective resolution of design conflicts become a core problem for effective decision making for PDD.

The description, representation, quantification, and transformation of product performance should be studied in the first place to implement PDD. At present, a unified terminology for product performance has not yet been available and there are many different definitions given in different fields of study and for different research subjects, for example, function requirements attribute [1], function-quality evaluation system [2], and the basic characteristics of the design process [3]. Compared with product function, product performance is more fuzzy and dynamic, which generally involves function, quality, structure, material, environment, and other factors related to the design, development, and use of a product. In this research, the study

of conflict coordination for PDD is based on the function-structure representation.

PDD is an innovative design method in which the realization of good function and the attainment of high quality are at the central place in the analysis, deliberation, design, calculation, and evaluation of design ideas. In the opinion of Wie et al. [4], PDD is an important part of the modern product design theory and methodology. Under the function-quality scheme, to meet customers' requirements on product performance, PDD specifically solves the problem of quantifying these requirements. In essence, PDD can be viewed as a process whereby product's performance characteristics are mapped to its structural characteristics. Researchers have done some research to facilitate the mapping process. Xie [5] proposed the method of generalized PDD which emphasizes the feedback-effect model and where the effects the product may have in the design process and the feedbacks it provides to its environment in response to the effects are modeled as performance knowledge. The knowledge can be used to describe the interactions between the whole process and its subprocesses, the interactions between the subprocesses, the interactions between the whole product and its parts, and the interactions between parts [6, 7]. In this sense, the focus of PDD is to establish the mapping process between product performance and product structure directly or indirectly. The mapping can be further used to construct the model to describe the correlation between performance parameters and structure characteristics and thus to analyze, evaluate, and coordinate design conflicts caused by different performance requirements throughout the design process.

In PDD, design conflicts are inevitable because all the performance requirements need be taken into account as a whole and, furthermore, the changes made to meet one performance requirement may affect how other requirements are met. Therefore, conflict resolution becomes a central problem in PDD. The conflict problems have been studied in design and the studies are mainly focused on the fields of concurrent design, collaborative design, axiomatic design, and multiobjective optimization. A number of conflict resolution methods have been developed and all of them have specific application areas. Firstly, the resolution methods [8–11] based on design rules reduction and design knowledge reasoning generally use design rules as the basis to implement the representation, retrieval, and reasoning of design knowledge. The main drawback of these rule-based methods is that it is difficult to develop a consistent and concise way of acquiring and representing design rules. Secondly, a number of methods have been developed to place conflict problems at the center [12–14], namely, the methods based on concurrent design, the methods based on case reasoning, the resolution strategy for layered conflicts, and the methods based on the feasibility analysis of resolution strategies. This type of conflict resolution methods can effectively solve conflict problems. However, in these methods, design conflicts are treated as being static as opposed to being dynamic and the propagation of changes made to improve product performance is overlooked.

The theory of invention problem solving (TRIZ) is an important technique for coordinating design conflicts, which offers guidelines for innovative design. It was proposed by Ahshuller's research team on the basis of analyzing 2.5 million patents around the world [15]. They made identified successful invention methods and rules from the solutions described in the patents and proposed to use these methods and rules to help designers in the solving of new problems. At present, there are a lot of scholars studying the conflict resolution method based on the TRIZ theory [16–20]. However, the application of TRIZ requires designers to have enough knowledge about TRIZ and be experienced in dealing with various design problems. In addition, the TRIZ theory aims to eliminate conflicts completely, which, under the uncertainty of the current market, seems to be impossible as the elimination a principal conflict may result in the occurring of other conflicts. Furthermore, TRIZ is used as a set of guidelines and rules that need to be understood by designers. This makes it difficult to use computer to implement automatic conflict resolution, and, as such, the cost of conflict resolution is high and the cycle for developing new technologies and applying them to real problems is long.

The Extension Theory (ET) is another important tool for conflict coordination, which emphasizes addressing the dynamic features of conflict problems by transforming the current solution space using various transformation methods. It was proposed by professor Cai et al. [21] in 1997 to formulate a model for exploring the possibility of the transformations of things and thus identifying the formal rules and methods for innovation. Later on, this method was used by researchers for solving conflict problems and so far it has been successfully applied in a number of areas such as new product development, decision making, business, and system control. It, together with TRIZ, has been studied by researchers to improve design innovation [22]. Zhao and Nan [23] applied ET to a number of design research areas such as conceptual design innovation based on ET, ET-based configuration design, and variant design and proposed an intelligent design method to solve conflict problem in design and thus provide a theoretical basis for innovative design, namely, ET-based design. ET-based design can effectively describe conflict problems and their formalized solving process, which solves dynamic conflict problems in design using various transformation methods [24]. Nevertheless, the application of the ET logic to quantitative solving and reasoning is still in the preliminary stage, which is subject to low efficiency and long development cycle.

In summary, the conflict problems in PDD are difficult to formulate, represent, convert, and resolve. In addition, the methods developed so far for resolving conflicts have various disadvantages such as being difficult to implement in a computer tool, involving a long cycle of development, and overlooking the dynamic features of conflicts. To address these disadvantages, this research aims to develop methods and tools that can facilitate designers' decision making by automatically resolving design conflicts. The ET-based design method has been chosen as a basis for this work as it offers a formalized high-level modeling scheme as well as both qualitative and quantitative reasoning methods. More importantly, the transformation methods used in ET-based design help

resolve dynamic design conflicts and attain system-level optimization. This paper describes a conflict resolution method based on the ET on the basis of understanding the nature of design conflicts, developing a formalized model for applying transformations to the design space to resolve conflicts, and utilizing the logical reasoning in the ET to identify and address change propagation caused by the transformations. The conflict resolution method is implemented in a prototype system and is successfully applied to the design of a screw air compressor (Figure 13).

2. Conflict Coordination Based on Propagation Analysis for PDD

When a design solution is changed to meet one specific performance requirement, the degrees this solution meets other performance requirements are also affected; that is, design conflicts arise. As customer-centered design becomes very popular, customers' performance requirements increasingly diversify. The diversification necessitates the study of conflict resolution for effectively supporting decision making in design and drives the development of PDD. Therefore, PDD has become an important part of the modern design research [2].

Product performance involves a set of parameters that are multidimensional, time varying, and highly coupled, and its quantification and calculation are complicated. This is more evident for the products that have many modules and thus the overall performance is very difficult to evaluate as all the modules may be subject to spatial and temporal changes. Taking the calculation of the noise of a screw air compressor as an example, every module in the compressor is actually a noise source and will influence the overall noise level. However, the overall noise cannot be simply evaluated by adding up every module's noise value because the spatial layout of the modules of, and the temporal interactions between, these modules need to be taken into account.

To address these challenges, this paper presents the development of a conflict resolution method based on the ET by studying the origin, formulation, and resolution of the conflicts between different performance requirements. The framework of this research is shown in Figure 1 and the pieces of work involved are described in detail as follows.

(1) *Formulation of the Conflict Problems in PDD.* Firstly, the requirements of product performance are taken as the main design target, and the measure for the distance between two cases is improved based on the distance definition given in the ET. This improved distance measure can be used to evaluate the similarity between the required performance and the performance attained by a specific product case. Thus, the most similar cases can be found from the base and the performance requirements each of these cases fails to meet can be identified. The conflict model is then constructed by analyzing the changes that need to be made to the attributes of the cases to meet the identified requirements. On the basis of this model, the classification of the core conflict problem can be realized, the propagation of the changes made to one performance requirement to others can be established, and the strategies for coordinating conflicts by utilizing transformation of the design space and propagation of changes can be obtained.

(2) *Coordination of Conflicts between the Core Performance (Which Represents the Performance Least Met by a Product Case throughout this Paper Unless Otherwise Stated) and the Product Structure Based on Propagation Analysis.* The structure of the most similar case is studied in detail and the substructure that affects the core performance mostly is defined to create a quantitative model for the core performance. The mapping between the quantitative descriptions of product's performances and its structure is researched. The propagation of design changes, when the transformations of the design space based on the ET theory are applied, is analyzed together with its influence on the performance attributes. Eventually, the attribute value of the core performance is calculated by taking into account various attributes collectively. On the basis of this value, the design solutions that meet the core performance requirements are selected.

(3) *Conflict Coordination for PDD Based on Change Analysis and Propagation.* A number of design schemes that meet customers' requirements are obtained by optimizing the core performance. Then, the attribute value of each scheme is calculated and the values before and after transformations are analyzed to obtain the threshold for a transformation and its resultant change propagation to take place. Next, the effect of propagation and the level of propagation are studied to calculate the degree of propagation (DOP) and obtain the classification knowledge (Table 5). Eventually, the dynamic transformation mechanism between product structure and product performance is established.

3. Formulation of the Conflict Problem for PDD

Using a case base in a design environment that requires fast response can shorten design cycle and improve design efficiency. Hence, in this paper, a design conflict resolution method based on transformation and propagation of product performance attributes is developed. Based on the improved distance measure in the ET (in the ET, this distance is termed the extension distance), the distance between the required performance and the performance a specific product case can attain is calculated. In this way, the number of the performance requirements that cannot be met is found out. Then, the changes of the attribute values of these performances in the case base are analyzed, and, thus, the number of the performances in conflicts and the core performance can be obtained. At last, the classification of conflict problems can be done and the coordination strategy of conflicts can be obtained.

3.1. Identifying the Unsatisfied Performance Requirements. The first step of the conflict resolution method is to define the performance that is not satisfied (Figure 11). Then, the conflict performance and its total number can be identified together with the correlations between the core performance and the

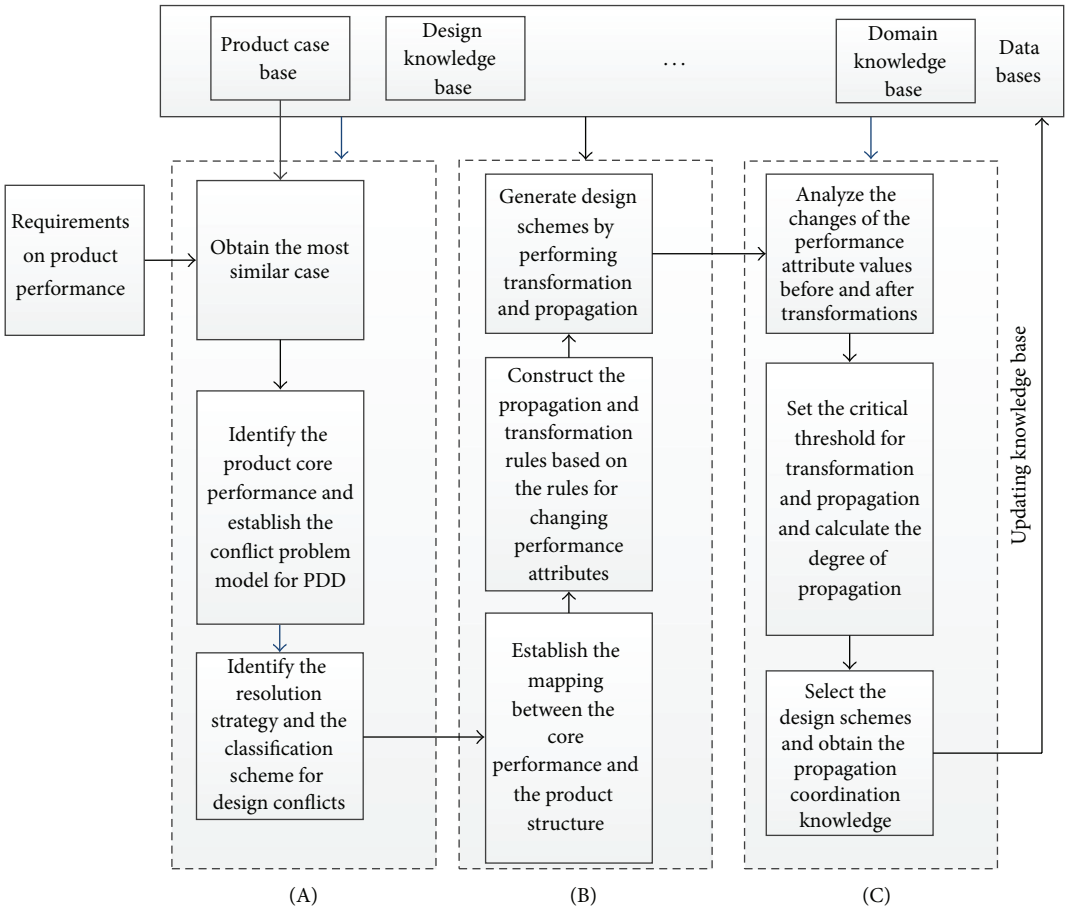


Figure 1: The framework for developing a conflict resolution method for PDD based on propagation transformation in the Extension Theory. (A) Defining the conflict problem in PDD, (B) conflict coordination between the core performance and the current product structure based on change propagation, and (C) calculation of product performance by taking into account change propagation and conflict coordination.

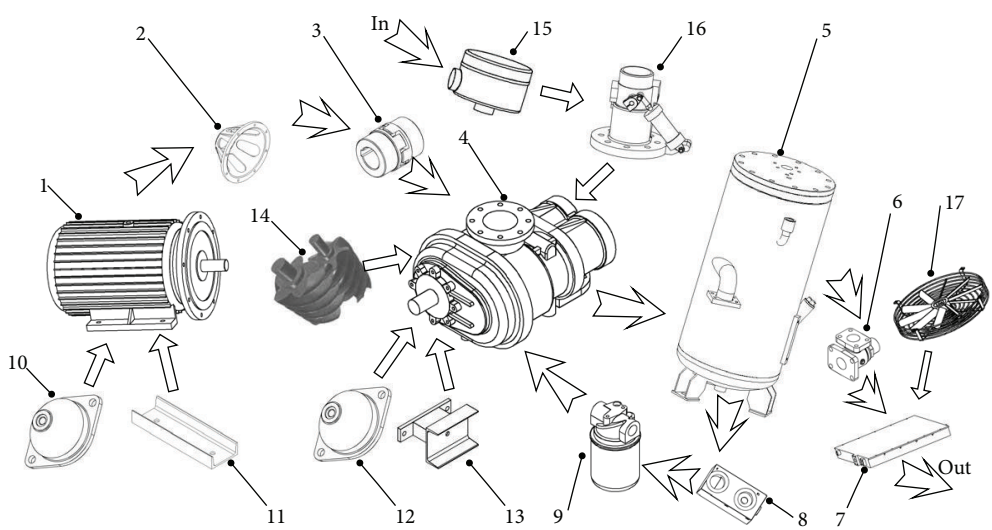


Figure 2: The main components and working schematic diagram of a screw air compressor.

Table 1: Partial data of different screw air compressors from the case base.

Model	Exhaust (m^3/min)	Exhaust pressure (Mpa)	Rated power (kw)	Noise (dB)	Weight (kg)	Fuel consumption (liter)	Stability
1-LG-1.45/10	1.45	1	11	70	500	18	Good
2-LG-15/7	15	0.7	90	90	1800	65	Best
3LG-1.2/13	1.2	1.3	11	70	500	18	Ordinary
4LG-13/7	13	0.7	75	90	1800	65	Better
5LG-2.2/7	2.2	0.7	15	70	550	18	Best
6LG-5.5/13	5.5	1.3	45	80	1100	35	Ordinary
7LG-3/7	3	0.7	18.5	74	650	20	Best
8LG-5.2/10	5.2	1	37	80	1050	35	Good
9LG-2.2/13	2.2	1.3	18.5	74	650	20	Ordinary
10LG-4.2/13	4.2	1.3	37	80	1050	35	Better
11LG-3.6/7	3.6	0.7	22	74	650	20	Better
12LG-7.2/7	7.2	0.7	45	80	1100	35	Best
13LG-1.9/10	1.9	1	15	70	550	18	Best
14LG-2.6/13	2.6	1.3	22	74	650	20	Ordinary
15LG-4.5/7	4.5	0.7	30	74	1000	35	Best
16LG-4/10	4	1	30	74	1000	35	Good
17LG-2.6/10	2.6	1	18.5	74	650	20	Good
18LG-3.4/13	3.4	1.3	30	74	1000	35	Ordinary
19LG-6/7	6	0.7	37	80	1050	35	Best
20LG-1.7/7	1.7	0.7	11	70	500	18	Better
21LG-6.3/10	6.3	1	45	80	1100	35	Good
22LG-10/7	10	0.7	55	82	1350	55	Best
23LG-1.7/13	1.7	1.3	15	70	550	18	Ordinary
24LG-8/10	8	1	55	82	1350	55	Better
25LG-3/10	3	1	22	74	650	20	Good
26LG-18/7	18	0.7	110	90	1800	65	Best

Table 2: The noise-reducing matter-element method for the screw air compressor.

Performance characteristic	Matter-element method	Configuration element set
Aerodynamic noise A_{11}	A_{111} reduces intake noise	Inlet strainer PE1 $\wedge$ inlet muffler PE3
	A_{114} reduces fan noise	Electrical machine PE5 $\wedge$ cooling fan PE6

others. The main measure for evaluating the degree a product meets customers' requirements is the distance between the required performance and the performance the product has attained. Since product performance generally involves multidimensional parameters with multiple units, the original data of performance need to be processed to remove the units; that is, the data is converted to values in the interval $[0, 1]$.

Actually, different extension distances describe different places from a point to an interval. Because the formula of the extension distance in [25] cannot ensure that values at the end of an interval are retrieved accurately, the right- and left-side extension distances need to be improved to address this issue, which results in the new formulas shown as follows.

The improved left-side distance applies to the situation that performance will be better when the performance value trends to be smaller in an interval $Y(y_1, y_2)$

$$d(x, Y) = \begin{cases} y_1 - x, & x \le \dfrac{y_1 + y_2}{2}, \\ x - \dfrac{y_1 + y_2}{2}, & x > \dfrac{y_1 + y_2}{2}. \end{cases} \tag{1}$$

The improved right-side distance applies to the situation that performance will be better when the performance value trends to be bigger in an interval $Y(y_1, y_2)$

$$d(x, Y) = \begin{cases} \dfrac{y_1 + y_2}{2} - x, & x < \dfrac{y_1 + y_2}{2}, \\ x - y_2, & x \ge \dfrac{y_1 + y_2}{2}. \end{cases} \tag{2}$$

Because product performance characteristics are not unique, the global similarity needs to be calculated by obtaining the values of all the local similarities collectively. The formula for the local similarity based on improved side distance is described as $V_{\text{pmd}}(x, Y) = 1 - |d(x, Y)|$, and the formula for the global similarity is described as Formula (3) where $V_{\text{pmd}}(x_j, Y_{ij})$ is the similarity of performance attribute between a case Y_{ij} and a design problem x_j, $R = \{x_1, x_2 \ldots, x_n\}$, and $E_i = \{Y_{i1}, Y_{i2}, \ldots, Y_{im}\}$ (the number of

Table 3: The muffler configuration schemes after the transformations for meeting the noise performance requirement.

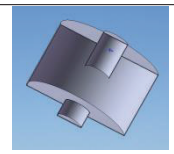
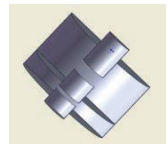
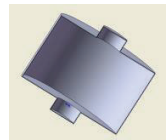
The new configuration elements of muffler	Configuration schemes	The muffler cutaway view cases after transformation	Noise reduction (dB)	Noise (dB)	Correlation degree
1	$T_4 \wedge T_1 \wedge T_1 \wedge T_1 \wedge T_1 \wedge T_2 \wedge T_2$		14.3	71.3	−0.058
2	$T_4 \wedge T_1 \wedge T_1 \wedge T_1 \wedge T_1 \wedge T_2 \wedge T_2 \wedge T_1 \wedge T_3$		18.1	68.5	0.088
3	$T_4 \wedge T_1 \wedge T_1 \wedge T_1 \wedge T_2$		12.1	74.2	−0.148

Table 4: Data about performance parameter of the screw air compressor before and after transformations.

Product	Noise (dB)	Intake pressure (Pa)	Rated power (kw)	Weight (kg)	Exhaust pressure (MPa)	Exhaust (m^3/min)	Fuel consumption (L)
0	80	100000	45	1100	1	6.3	35
1	71.3	99697.3	45	1124.3	1	6.28	35
2	71.3	99700.1	45	1200.8	1	6.29	35
3	68.5	99386.9	45	1130.8	1	6.24	35
4	68.5	99390.9	45	1215.1	1	6.22	35
5	74.2	99690.5	45	1122.3	1	6.27	35
6	74.2	99693.6	45	1183.3	1	6.28	35

variables in a design problem is n and the number of variables in a case is m, $m > n$). The weight of performance attribute w_j is defined through Analytic Hierarchy Process (AHP) and $\max V_{\text{amd}}(R, E_i)$ is outputted according to

$$V_{\text{amd}}(R, E_i) = \sum_{j=1}^{n} w_j V_{\text{pmd}}(x_j, Y_{ij}). \tag{3}$$

Assume $\exists V_{\text{pmd}}(x_\varnothing, Y_t) = 1$, $i < t < j$; namely, the similarity between a product case Y_t and an unsatisfied design performance characteristic $x_\varnothing$ is 1. According to the value of $\max V_{\text{amd}}(R, E_i)$, the number of cases with $V_{\text{pmd}}(x_j, Y_{ij}) < 1$ and its corresponding performance P_j can be defined. Then, the set of performances for which customers' requirements are not met can be described as follows:

$$S_{P_{nm}} = \left\{P_{nm}^j \mid V_{\text{pmd}}\left(v\left(P_{nm}^j\right), Y_{ij}\right) < 1, \; x_j = v\left(P_{nm}^j\right), j = 1, 2, \ldots, n\right\}. \tag{4}$$

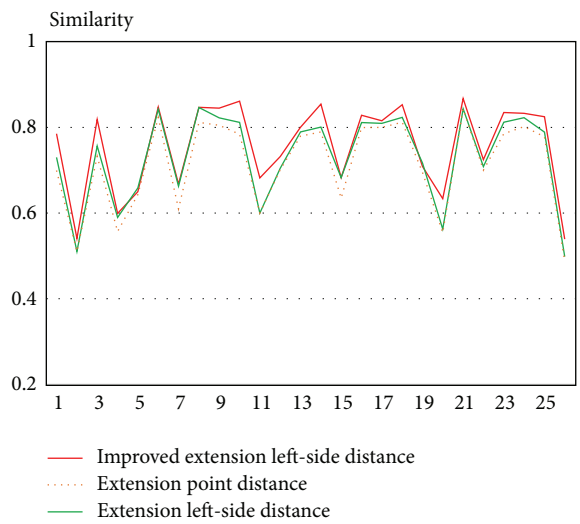
3.2. Modeling the Product Performance Conflict Problem. Since PDD involves multiple types of performance, the optimization of product performance is a multiple-objective optimization problem. To establish the conflict resolution model, quantitative description of the performance requirements needs to be obtained in an accurate way. Hence, in this work, the correlations between different performances are identified by analyzing the evolution of performance attributes for all the cases in the product case base. Assume that there are n_1 product cases, each of which is marked as PC_i where $i = 1, \ldots, n_1$, and there are n_2 unsatisfied performances, each of which is marked as P_{nm}^{ij}, where $j = 1, \ldots, n_2$. Finally, the performance instance P_{nm}^{ij} that has the minimum similarity with the required performance $\min V_{\text{pmd}}(x, Y)$ is selected as the core performance. If the product cases are ranked in a descending order in terms of their performance attribute values that are marked as $v(P_{nm}^{ij})$, then a design matrix about PC_i and P_{nm}^{ij} can be constructed as follows:

$$DM = \begin{array}{c} \\ PC_1 \\ \vdots \\ PC_{n_1} \end{array} \begin{array}{c} \begin{array}{ccc} P_{nm}^{i1} & \cdots & P_{nm}^{in_2} \end{array} \\ \begin{bmatrix} v_{11} & \cdots & v_{1n_2} \\ \vdots & \ddots & \vdots \\ v_{n_1 1} & \cdots & v_{n_1 n_2} \end{bmatrix} \end{array}. \tag{5}$$

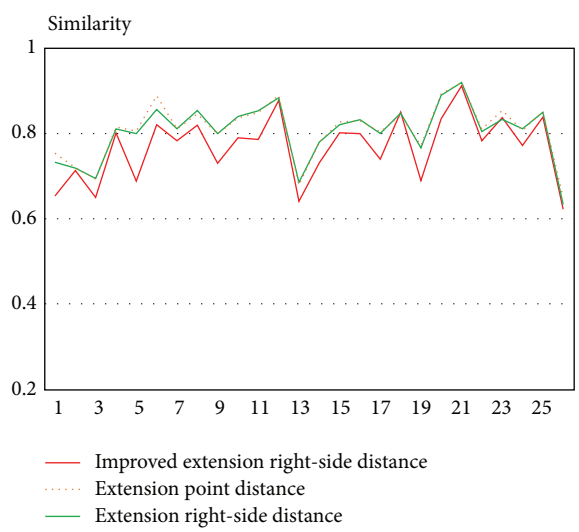
Since each column is ranked in a descending order, then we have $v_{11} > v_{21} > \cdots > v_{n_1 1}$. Hence, this design matrix can

Table 5: Degree of propagation of the screw air compressor propagation properties.

First-order propagation	Noise (dB)	Intake pressure (Pa)	Weight (kg)	
Quadratic propagation				Exhaust (m^3/min)
γ_1	0.045	1.566	0.126	6.6×10^{-5}
γ_2	0.051	1.772	0.537	3.3×10^{-5}
γ_3	0.071	3.804	0.191	11.42×10^{-5}
γ_4	0.080	4.254	0.804	13.13×10^{-5}
γ_5	0.025	1.355	0.098	9.7×10^{-5}
γ_6	0.029	1.544	0.420	6.5×10^{-5}



(a)



(b)

Figure 3: A comparison of retrieval performances before and after using the improved distance. (a) Comparison of the similarities between a point and an interval in terms of exhaust before and after the improved distance is used. (b) Comparison of the similarities between a point and an interval in terms of weight before and after the improved distance is used.

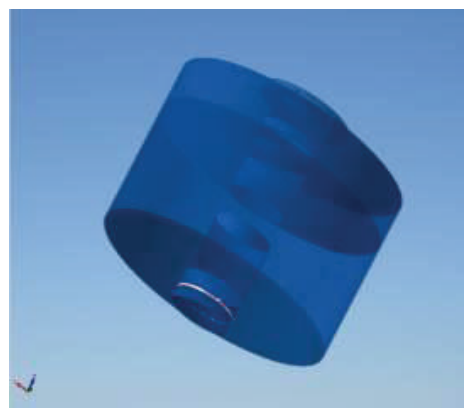


Figure 4: 3D model of the expanded muffler.

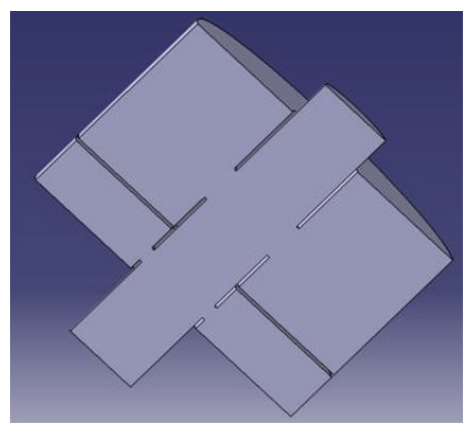


Figure 5: Cutaway view of the expanded muffler computational domain.

be calculated and converted to an upper triangular matrix as follows:

$$DM' = \begin{array}{c} \\ PC_1 \\ \vdots \\ PC_{n_1} \end{array} \begin{array}{c} \begin{array}{ccc} P_{nm}^{i1} & \cdots & P_{nm}^{in_2} \end{array} \\ \begin{bmatrix} v'_{11} & \cdots & v'_{1n_2} \\ 0 & \ddots & \vdots \\ 0 & 0 & v'_{n_1n_2} \end{bmatrix} \end{array}. \quad (6)$$

According to the above formula, we can get the conclusion that any performance attribute value can be expressed by the others' value in the matrix; for example, $P_{nm}^{ij} = P_{nm}^{1j} + P_{nm}^{2j} -$

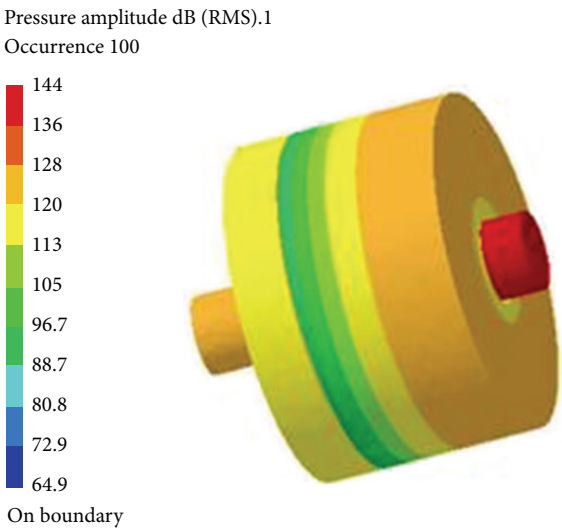


Figure 6: The picture of sound cloud of muffler.

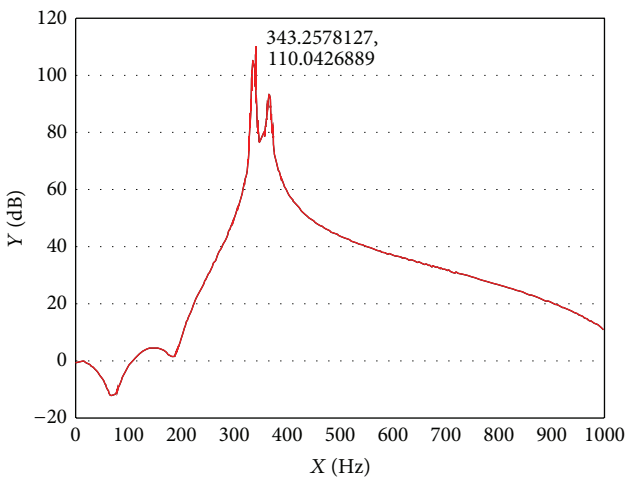


Figure 7: Transmission loss curve of pressure muffler.

Figure 8: Extension design prototype system.

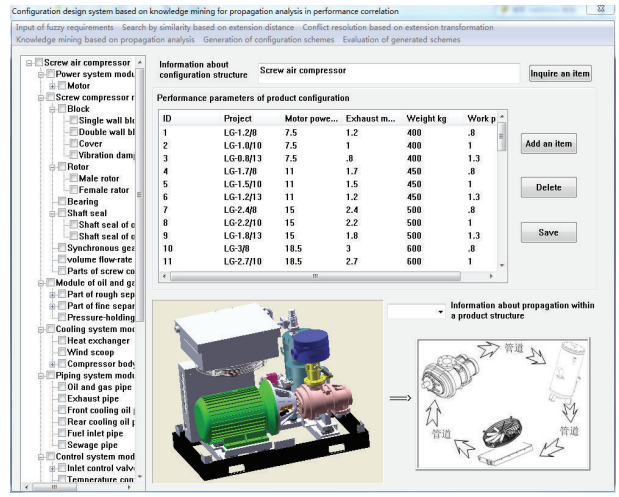


Figure 9: Graphical user interface of conflict coordination design system.

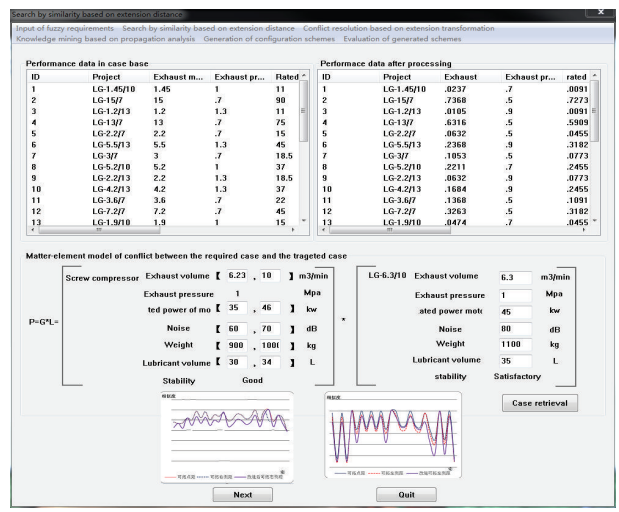


Figure 10: Performance requirements and similarities calculation.

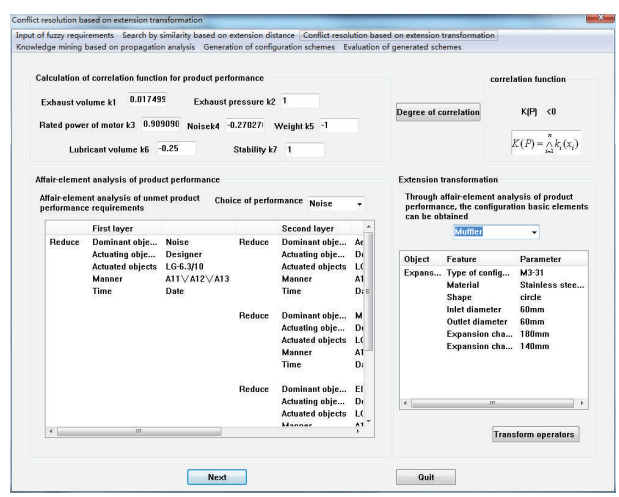


Figure 11: The conflict resolution module.

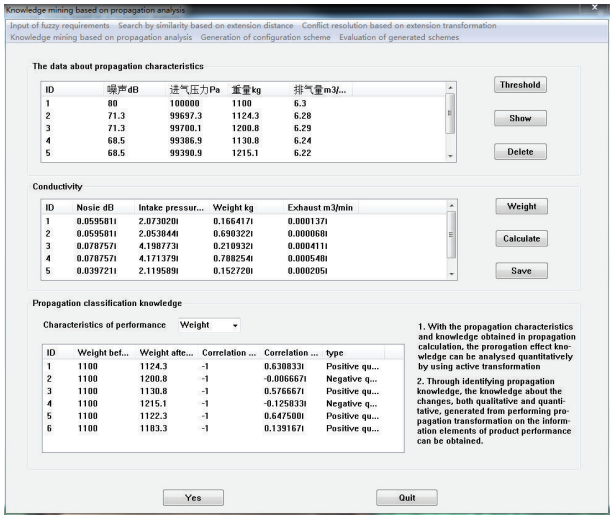


Figure 12: Propagation knowledge extraction.

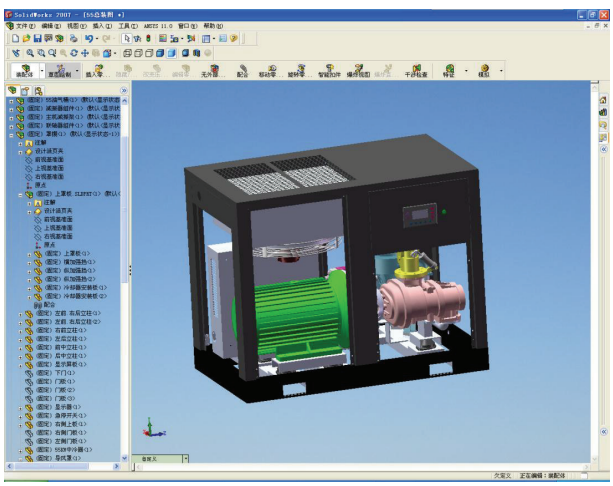


Figure 13: 3D model of the screw air compressor.

$P_{nm}^{3j} + \cdots - P_{nm}^{n_1 j}$. If $j = 1, \ldots, n_2$ is taken into the formula, the changes of other performance can be obtained when $v(P_{nm}^{ij})$ is ranked in a descending order. The signs of variables in this formula can indicate the correlations between different performance requirements; that is, a positive sign means that there are no conflicts between two performances, a negative sign means that there are conflicts, and a missing item means there is no correlation between two performances.

Therefore, a conclusion can be drawn that the overall performance requirements can eventually be met if the conflicts between the core performance and those performance requirements with negative signs in the matrix during the coordination process can be resolved. Assume that there are n_3 performances with positive signs; then the product performance conflict model can be described using the following representation, where the sign $*$ means the correlation between two variables as opposed to a logical operator:

$$Q = \left(P_{nm}^{ij} \wedge P_{nm}^{i1} \wedge \cdots \wedge P_{nm}^{in_3}\right) * CP_i. \quad (7)$$

For multiperformance optimization, the similarities between product cases are all different, the correlations between these performances are fuzzy and difficult to be determined, and an integrated quantitative mapping from performance to structure is lacking (Figure 10). When optimizing performance attributes by changing the structure parameter values, it is difficult to determine the structure parameters a specific performance attribute is related to, and, moreover, this kind of problem is subject to high complexity and low efficiency while it is difficult to implement system-level optimization and control the propagation of design changes (Figure 12). Currently, optimization for the single-performance problem mainly relies on improving the structure related to the performance according to designers' experiences.

Therefore, in this paper, the coordination process for conflict problem is divided into subprocesses. At first, conflict problem between the core performance and product structure is resolved. Next, the conflict problem in overall performance is coordinated to generate design schemes which meet customers' requirements by performing transformations and utilizing propagations.

4. Coordination of the Conflicts between the Core Performance and Product Cases Using Propagation Transformation

The essence of meeting the core performance is quantification of the correlations between performance attributes and product structure parameters and formulates them. The basic ways to solve the conflict problem in the ET-based design include (1) changing the product structure parameters, (2) changing the values of product performance attributes, and (3) changing both the structure parameters and performance attributes. Using the ET-based design method can give a formal description to the multifactor conflict problem as well as a solution process whereby conflict problems are analyzed in a dynamic way and using the combination of quantitative and qualitative methods.

4.1. Formulation of the Conflict between the Core Performance and Product Structure. Based on the conflict model of product performance, the model that describes the conflict between the core performance and product structure can be expressed as $Q_1 = P_{nm}^{ij} * CP_i$. In this expression, the performance and the product case in solving a conflict problem are given, but it could not explicitly express issues such as the correlation between the core performance and product structure and the structure parameters that most influence a specific performance attribute. Therefore, in order to describe the problem more clearly, the model should be improved by adding quantitative operations.

The core performance optimization must be implemented by representing product structure parameters quantitatively as well as by developing mathematical expressions for the calculation of performance attributes and product structures. While the number of the product structure performance is not unique and the effects on the product

performance are different, the core performance structure should be used as the reference for performing propagation transformation to resolve conflict. The structure for a performance specifically refers to the minimum components that can attain a required function at an acceptable level of satisfaction.

Use P_{nm}^{ij} to represent l product structures $PS_1, PS_2, \ldots, PS_l$ and use $PSP_1, PSP_2, \ldots, PSP_l$ to represent the structure performance, so the calculation of P_{nm}^{ij} can be expressed as $v(P_{nm}^{ij}) \rightarrow v(P^i) = f[v(PSP_1), v(PSP_1), \ldots, v(PSP_l)]$, where the sign $\rightarrow$ means tend to optimization. In order to achieve the optimization of the core performance quickly, the structure performance which has the most influence on the performance should be selected as the object of transformation and propagation; that is, $CP_i = \{PS_e \mid PS_e = \max R(P_{nm}^{ij}, PSP_e),\ e = 1, \ldots l\}$.

Thus, the quantified conflict problem model can be expressed as $Q_{11} = [v(P_{nm}^{ij}) \rightarrow v(P^i)] * Z_{PS_e}$, where $[v(P_{nm}^{ij}) \rightarrow v(P^i)]$ means achieving the performance demand parameter goal, $v(P^i)$ means product case performance parameter and Z_{PS_e} means the eth-associated structure element (Table 4). The element theory in the ET uses the formalized three-tuple representation $B = (O, C, V)$ to describe a matter, an affair, or a relation where O means an object, C means the set of characteristics of object O, and V is the set of values of the characteristics. When describing a complex problem, a compound element representation is used to reduce the model dimension, being expressed as the combination of the matter-element, the affair-element, and the relation-element [26]. PS_e means the establishment of the relationship chain of the parts in a product based on the energy transfer path or power transfer path when the product is in normal work condition. Assume that a product has s parts; then it can be expressed as $PS_{e1}, PS_{e2}, \ldots, PS_{es}$; thus the conflict problem model can be further quantified to the parts layer

$$Q_{111} = \left[v\left(P_{nm}^{ij}\right) \longrightarrow v\left(P^i\right)\right] * \left\{Z_{PS_{ei}}\right\}_{i=1\sim s}. \tag{8}$$

4.2. The Propagation Process for the Conflict Coordination Method. The propagation process for conflict coordination based on transformations is to create propagation rules which are based on the parts relationship chain. In process, conflict problems are rationalized based on propagation transformation using the ubiquitous correlation and implication principle among the objects in a conflict problem. Propagation reasoning thus is the propagation transformation ($_{\varphi}T$) operation method caused by the active transformation (φ).

An arbitrary component element $Z_{PS_{ei}}$ is selected as the active transformation object, so that we can get $\varphi v[c(Z_{PS_{ei}})] = v'[c(Z_{PS_{ei}})]$. This will generate two kinds of propagation processes and their respective propagation rules are summarized in the following formal representations. The setting, priority, and selection of, and the weight assignment for, different transformation types are beyond the scope of this paper and can be found in the literature [25].

(1) The propagation reasoning based on component object's own features and values is as follows:

propagation rule 1: propagation transformation based on the object:

$$\begin{aligned}\left({}_{\varphi}T_j Z_{PS_{ei}} = Z'_{PS_{ei}}\right) &\vDash \left({}_{Z_{PS_{ei}}}T_j(c, v) = \left(c', v'\right)\right)\\ &\vDash k'\left(c'\left(Z'_{MP_{xi}}\right)\right);\end{aligned} \tag{9}$$

if $k' < 0$, then $j = j + 1$; do ${}_jT_{j+1}$, until $k' > 0$;

propagation rule 2: propagation transformation based on the feature values:

$$\begin{aligned}&\left({}_{\varphi}T_j\, c_t\left(Z_{PS_{ei}}\right) = c'_t\left(Z_{PS_{ei}}\right)\right)\\ &\vDash \left({}_{c_t(Z_{PS_{ei}})}T v_t = v'_t\right)\\ &\vDash {}_{v'_t}T\left(\{c_t\}, \{v_t(c_t)\}\right) = \left(\{c'_t\}, \{v'_t(c'_t)\}\right)\\ &\vDash k\left(\left\{c'\left(Z_{PS_{ei}}\right)\right\}\right);\end{aligned}$$

if $k' < 0$, then $j = j + 1$, $t = t + 1$; do ${}_jT_{j+1}^{t+1}$, until $k' > 0$. (10)

(2) The propagation reasoning based on component object's own features and relationship chain is as follows:

propagation rule 3:

$$\begin{aligned}\left({}_{\varphi}\{T\} Z_{MP_{ei}} = Z'_{MP_{ei}}\right) &\vDash \left({}_{Z_{MP_{ei}}}TZ_{MP_{ej}} = Z'_{MP_{ej}}\right)_{j \neq i \in s}\\ &\vDash K\left(Z'_{MP_{ej}}\right) = k\left[f\left(c\left(Z'_{MP_{ej}}\right)\right)\right];\end{aligned}$$

if $k_t < 0$, then $\varphi = \varphi_{k_t<0}^{t+1}$; do ${}_{\varphi_{k_t>0}^{t+1}}\left\{T^{t+1}\right\}$, until $k_{t+1} > 0$. (11)

$\varphi_{k_t<0}^{t+1}$ means the $(t + 1)$th-order active transformation values which are determined by the tth-order propagation transformation, $t = 0 \sim n$.

Then the performance and product structure conflict based on the propagation rules can be quantitatively expressed as

$$\begin{aligned}(\varphi, T)\, Q &= (\varphi, T)\left\{\left[v\left(P_{nm}^{ij}\right) \longrightarrow v\left(P^{i}\right)\right] * \left\{Z_{PS_{ei}}\right\}_{i=1\sim s}\right\} \\ &= (\varphi, T)\left[v\left(P_{nm}^{ij}\right) \longrightarrow v\left(P^{i}\right)\right] \\ &\quad * (\varphi, T)\left\{Z_{PS_{ei}}\right\}_{i=1\sim s} \\ &= \left[f\left\{v\left(PSP_{e}\right)\right\} \overset{(\varphi,T)}{\longrightarrow} v\left(P^{i}\right)\right] \\ &\quad * \varphi Z_{PS_{ei}} \wedge {}_{\varphi}\{T\}\left\{Z_{PS_{ej}}\right\}_{j\neq i} \\ &= \left[f\left\{v\left(PSP_{e}\right)\right\} \overset{(\varphi,T)}{\longrightarrow} v\left(P^{i}\right)\right] \\ &\quad * \left[\varphi, {}_{\varphi}\{T\}\right] \cdot \left[Z_{PSi}, \ldots, Z_{PS_{es}}\right]^{TM} \\ &= k'\left[f\left(v\left[cPS_{ei}\right]\right), \ldots, \left(v\left[cPS_{es}\right]\right)\right]_{(\varphi, {}_{\varphi}\{T\})\cup v(P^{i})}.\end{aligned} \tag{12}$$

In this expression, TM means transposition; $f(\cdot)$ means the integration function; and $k(\cdot)$ means the correlation function. The correlation function is at the center place in the solving of conflict problem using the ET theory, which is constructed using the extension distance. In this sense, only when $k(\cdot) > 0$ can a conflict problem be coordinated.

The algorithm for resolving conflicts between performance attributes and structure parameters using propagation transformation is explained as follows.

Step 1. For $\{Z_{PS_{ei}}\}_{i=1\sim s}$, a group φ is given arbitrarily. Perform active transformation on the feature and then generate a group of propagation transformations ${}_{\varphi}\{T\}$ according to the propagation reasoning rules discussed above.

Step 2. For the transformation performed, evaluate the propagation effect, the degree of propagation, and the correlation formula [27]; calculate the value for each function so as to identify the direction of the transformation.

Step 3. Evaluate the feasibility of the propagation transformations performed; calculate the values of $v(PS_e)$ and $v(P_{nm}^{ij})$; find out if the condition $k'[f(v[c(PS_{ei})], \ldots, v[c(PS_{es})])]_{(\varphi, {}_{\varphi}\{T\})\cup v(P^{i})}$ is met; if it is met, then output the scheme as a potential solution and jump to Step 5; otherwise, jump to Step 4.

Step 4. Perform multiple-order propagation transformation; perform the assignments of $\varphi = \varphi_{k_t<0}^{t+1}$ and ${}_{\varphi_{k_t<0}^{t+1}}\{T^{t+1}\}$ based on the results obtained from the last transformation; assign $t = t++$, $t = 0 \sim n$; jump to Step 1.

Step 5. End the process; output the set of feasible propagation coordination schemes.

Through the coordination of the conflicts between performance attributes and structure parameters, a variety of product structure transformation design schemes are obtained as well as the knowledge about performing propagation transformation to meet performance requirements. However, the overall satisfaction degree of product performance also needs to be evaluated even though a single-performance attribute is met. Thus, the feasibility, advantages, and disadvantages of the design schemes generated after the propagation transformations can be further evaluated to support designers to make informed decisions.

5. The Coordination of Design Conflicts Based on Transformation of Product Structure and Propagation Analysis

In the PDD, when some of customers' performances requirements are met, the structure parameters that are correlated with these requirements will be changed to achieve improved satisfaction. However, currently, design engineers mostly change the structure of a product using their experiences firstly and then test the resultant design to see whether some performance requirements are better met. This process greatly relies on the expertise of engineers and generally leads to longer development time so that customers' new requirements cannot be met in a swift way. In addition, this method pays little attention to analyze and use the data generated after the transformations are performed, leading to the situation that the influences of these transformations on other performances are overlooked and the correlations between the data are also not utilized properly. Knowledge mining based on propagation analysis is a data mining method that is specifically used to resolve conflict problems. This method can help designers make informed decisions in the design process through analyzing the correlations between performance attributes and structure parameters before and after transformations are performed and thus providing new knowledge on the resolving of conflict problems.

5.1. The Determination of Propagation Properties Based on the Designing Schemes Generated after Transformations. Through the transformation of a product structure, an expected performance value can be achieved, while the transformation may result in changes to other performances. The inherent relationship between the performances and the change of performances in the product instance can be obtained, but it is hard to determine where these changes come from and whether these changes happen actively or are the results of the propagation process. The conflict model for all the performances can be represented as

$$\begin{aligned}Q_2 = &\left(v\left(P_{nm}^{ij\prime}\right) \wedge v\left(P_{nm}^{i1}\right) \wedge \cdots \wedge v\left(P_{nm}^{in_2}\right)\right) \\ &* \left(c\left(Z'_{PC_i}\right) \cup v\left[c\left(Z'_{PC_i}\right)\right]\right).\end{aligned} \tag{13}$$

Other performance values of the product when its core performance is satisfied can be calculated after the change of product's structure. The determination of propagation performance is based on two parameters, namely, the propagation

effect and the propagation sensitivity. The latter parameter ϑ means that other performance parameters have a change and the range of the change exceeds the threshold value of the propagation effect. The former refers to the d-value of the product performance parameters before and after the transformation; namely, $\Delta P_i = |P_i - P_i'|$. Then, the d-value of all products performance before and after the transformation can be calculated; only if $\Delta P_i > \vartheta > 0$, this performance is called the propagation object. On this basis, according to the propagation process, the level of propagation performance can be determined.

The degree of propagation is used to reflect the degree that changes to one product's performance parameters are propagated to others in the process of the propagation. The bigger the degree of propagation, the more obvious the propagation effect expression and it will the higher the correlation between two performances reflect. To resolve the problems that involve multiple active transformations and multiple-level propagations, the integrated degree of propagation can be constructed as follows:

$$\gamma = \frac{\left|P^i - P^{i\prime}\right|}{\sum \lambda_i \left\{\Delta v\left(c\left(PS_{ei}\right)\right)\right\}}. \tag{14}$$

5.2. The Classification Method for the Performance Propagation Knowledge. The extension set in the ET theory is an important tool of the classification of the propagation knowledge, which includes the basic element $Z_{PS_{ei}}$ and the correlation function of performance attributes values $k[f(v[c(PS_{ei})], \ldots, v[c(PS_{es})])]$. And, after finishing a group of transformations, the extension correlation function can be expressed as $k'[f(v[c(PS_{ei})], \ldots, v[c(PS_{es})])]_{(\varphi, {}_{\varphi}\{T\}) \cup v(P^i)}$. Then the extension set can be expressed as

$$\begin{aligned} \widetilde{E} = \Big\{ & \left(Z_{PS_{ei}}, h, h'\right) \mid Z_{PS_{ei}} \in \left\{Z_{PC_j}\right\}, \\ & h = k\left[f\left(v\left[c\left(PS_{ei}\right)\right], \ldots, v\left[c\left(PS_{es}\right)\right]\right)\right], \\ & h' = k'\left[f\left(v\left[c\left(PS_{ei}\right)\right], \ldots, v\left[c\left(PS_{es}\right)\right]\right)\right]_{(\varphi, {}_{\varphi}\{T\}) \cup v(P^i)} \Big\}. \end{aligned} \tag{15}$$

The greatest characteristic of the classification based on the extension set is that it can deal with dynamic data. Existing classification methods are mostly focused on static data and fail to take the classification of the performance attributes into account after transformation. In this paper, the focus is on identifying the dynamic propagation knowledge caused by transformation. The purpose of the classification is to obtain the following transformation types:

$$\begin{aligned} & \left[E_+^i \longrightarrow \dot{E}_+^i\left({}_{P_j}TP_i\right)\right] \cup \left(P_i \longrightarrow P_i'\right) \\ & E_-^i \longrightarrow \dot{E}_+^i\left({}_{P_j}TP_i\right) \\ & E_0^i \longrightarrow \dot{E}_+^i\left({}_{P_j}TP_i\right) \\ & E_0^i = \dot{E}_0^i\left({}_{P_j}TP_i\right) \cup \left(P_i = P_i'\right). \end{aligned} \tag{16}$$

In the above expression, E_+^i, E_-^i, E_0^i, $\dot{E}_+^i({}_{P_j}TP_i)$, and $\dot{E}_0^i({}_{P_j}TP_i)$ represent the positive domain, the negative domain, the zero frontiers, the qualitative change domain, and the extension frontier of the performance parameter, respectively.

5.3. The Algorithm for Performance Conflict Coordination Based on Propagation Analysis for Product Configuration Design

Step 1. Find a product case that matches the expected requirement and obtain the performance data $v\{P^{ij}\}$ of the case; then obtain the product performance data set $\{v\{P^{ij}\}, v\{P^{ij\prime}\}\}$ after a variety of transformations $(\varphi, {}_{\varphi}\{T\})$ of the case's structure.

Step 2. Calculate $\Delta v(P^{ij})$, the differences between the performance data set before and after a transformation, and judge the propagation properties by the given threshold δ_{ij}.

Step 3. Obtain the information of propagation performance properties that changed before and after transformation; then calculate the degree of propagation $\gamma = |P^i - P^{i\prime}| / \sum \lambda_i \{\Delta v(c(PS_{ei}))\}$ of the propagation properties caused by all the active transformations, to judge the degree of changes for other performances.

Step 4. Extract the knowledge of the propagation transformation; obtain the design schemes generated after the transformation and propagation using positive qualitative change and positive quantitative change based on the extension set $\widetilde{E}$.

Step 5. Extract the rules for transformation and propagation and store them in the rule base.

6. A Case Study

Compressed air is the second largest energy after electricity, as well as being used as a gas source in industrial processes with multiple usages. Screw air compressor, as a power system utilizing the compressed air power, is used in many industrial sectors such as petroleum, chemical engineering, electric power generation, mechanical engineering, the textile industry, automotive engineering, food, medicine, biochemical, and defense. A screw air compressor is mainly composed of the motor oil-gas separate barrel, the cooling system, the air-conditioning system, the lubricating system, the safety valve, and the control system. The structure of a particular compressor studied in this research is shown in Figure 2. As shown in the figure, it consists of the following components: 1: 55 kw motor (10: absorber component; 11: motor vibration damper); 2: center rest; 3: couping component; 4: head assembly (12: absorber component, 13: host damping frame, 14: male and female rotors, 15: air filter assembly, and 16: inlet valve component); 5: oil and gas separator component; 6: minimum pressure valve; 7: intercooler; 8: thermostat valve; 9: oil filter component; 17: axial flow fan assembly.

The improved left-side distance is evaluated by calculating the similarities (shown in Figure 3(a)) between the partial cases in the case base (detailed data shown in Table 1) and a product with an expected exhaust value (between 6 m^3/min and 10 m^3/min). The same evaluation has also been done for the right-side distance in the retrieval for a compressor with a weight between 550 kg and 650 kg, with detailed comparison of case retrieval results shown in Figure 3(b). By comparing the similarities calculated using different methods (extension point distance, left-side distance and right-side distance, improved left-side distance, and improved right-side distance) for the retrieval, it is shown that the improved left-side and right-side distances achieve better retrieval precision than other methods.

(1) *The Determination of Conflict Problem in the Product's Performance.* The performance requirement of the screw air compressor can be expressed using the model as follows:

$$\begin{bmatrix} \text{screw air compressor} & \text{exhaust} & [6, 10]\ m^3/\text{min} \\ & \text{exhaust pressure} & 1\ \text{Mpa} \\ & \text{rated power} & [17, 23]\ \text{kw} \\ & \text{noise} & [60, 70]\ \text{dB} \\ & \text{weight} & [550, 650]\ \text{kg} \\ & \text{fuel consumption} & [19, 24]\ \text{L} \\ & \text{stability} & [9.6, 10.4] \end{bmatrix}. \tag{17}$$

Partial data about different screw air compressors from the case base is shown in Table 1, which is searched through for similar cases using the methods discussed above.

The similar matrix is constructed with the numbers obtained using the formulas for local similarity:

$$s = \begin{bmatrix} 0.6553 & 1 & 0.9182 & 0.975 & 0.9262 & 0.9184 & 0.95 \\ 0.7369 & 0.8 & 0.3636 & 0.725 & 0.1144 & 0.1224 & 0.8833 \\ 0.6421 & 0.8 & 0.9182 & 0.975 & 0.9262 & 0.9184 & 0.7166 \\ 0.8421 & 0.8 & 0.5 & 0.725 & 0.1144 & 0.1224 & 0.8833 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0.8947 & 1 & 0.6818 & 0.875 & 0.4465 & 0.3265 & 0.8833 \\ 0.7369 & 1 & 0.9818 & 0.935 & 0.9631 & 0.9592 & 0.95 \\ 0.579 & 0.8 & 0.1818 & 0.725 & 0.1144 & 0.1224 & 0.8833 \end{bmatrix}. \tag{18}$$

The weights of different performances are calculated using the AHP method:

$$w = [0.3691, 0.2417, 0.0675, 0.1334, 0.0406, 0.0267, 0.1212]. \tag{19}$$

The overall similarity between customers' performance requirements for a screw air compressor and performance attributes of the compressors in the case base is calculated according to the formula of global similarity which is represented as $\text{sim}(R, E_i) = \sum_{j=1}^{n} w_j \text{sim}(R_i, E_i)$:

$$\begin{aligned} \text{sim}(R, E_i) = [\ & 0.7850\ \ 0.5406\ \ 0.8183\ \ 0.5992\ \ 0.6499\ \ 0.8476\ \ 0.6676\ \ 0.8467 \\ & 0.8448\ \ 0.8613\ \ 0.6820\ \ 0.7319\ \ 0.8001\ \ 0.8544\ \ 0.6828\ \ 0.8281\ \ 0.8153 \\ & 0.8527\ \ 0.7087\ \ 0.6336\ \ 0.8673\ \ 0.7245\ \ 0.8346\ \ 0.8327\ \ 0.8249\ \ 0.4542\ ]^T. \end{aligned} \tag{20}$$

According to the calculation result, the product case (LG-6.3/10) will be selected for its max $V_{\text{amd}}(R, E_i)$ value.

(2) *Conflict Coordination Using Transformation and Propagation.* Noise is identified as the performance attribute to be optimized after analyzing the performance attribute values of the most similar case and the values of the required performance. The model for describing the conflict between product performance and product structure can be described as follows:

$$Q = G * L = [P(80 \longrightarrow 40)] * \begin{bmatrix} \text{Expansion type silencer} & \text{Material} & \text{Corrosion resistant plate} \\ & \text{Shape} & \text{Circle} \\ & \text{The diameter of entrance} & 60\ \text{mm} \\ & \text{Outlet diameter} & 60\ \text{mm} \\ & \text{Expansion type diameter} & 180\ \text{mm} \\ & \text{Expansion type length} & 140\ \text{mm} \end{bmatrix}. \tag{21}$$

The correlation model for calculating the values of noise and structure properties of the screw air compressor is established according to resistance muffler which belongs to low-frequency noise. The noise reduction formula of resistance muffler can be calculated using the method discussed in literature [28]. The most similar case does not meet the noise requirements and does not have an inlet muffler mounted. Since the fundamental frequency of different type of inlets is different and the most similar case's fundamental frequency is 99 MHz, there are no mufflers in the case base that can meet the requirement. Thus, the method based on transformation and change propagation is used to resolve the conflict. In this case, the conflict problem is resolved using several transformations by several main matter-element paths after

comparing several aerodynamic noise characteristics and the noise value, as shown in Table 2.

The structure of muffler after transformation is got and shown as follows by using the propagation coordination algorithm:

$$\left(t_{29}^{2} \wedge t_{110}\right) l' = \begin{bmatrix} \text{Expansion type silencer} & \text{Material} & \text{Glass steel} \vee \text{galvanized steel sheet} \\ & \text{Shape} & \text{Circle} \\ & \text{Entrance diameter} & 85\,\text{mm} \\ & \text{Exit diameter} & 85\,\text{mm} \\ & \text{Expansion type diameter} & 340\,\text{mm} \\ & \text{The first expansion chamber} & 161.5 \\ & \text{The second expansion chamber} & 58.4 \end{bmatrix}. \quad (22)$$

Currently, there are no accurate calculations methods for machine noise; the overall noise cannot be obtained by simply adding all the noise sources. Assume that the noise intensities of k noise sources are $n_1, n_2, \ldots, n_k$; the overall unit noise z_n is $z_n = \log(10^{n_1} + 10^{n_1} + \cdots + 10^{n_k})$. Actually some measures to reduce noise are taken; the overall unit of actual noise f when the noise reduction is considered is the difference of the total noise and noise reduction sound intensity and can be described as $z_r : z = z_n - z_r$. The design schemes generated to meet the noise performance are shown in Table 3.

The interface between the muffler and other components is not changed during the propagation process, and, as such, other models remain unchanged.

(3) *Knowledge Mining for the Conflict Resolution Process.* Data about the performance attributes of the compressor after transformations is obtained through both theoretical calculations and simulation, which are added to the performance database after the transformations. For example, the formula of pressure loss due to friction between the inner wall of muffler and airflow is calculated as follows:

$$\Delta P_f = 10\xi_l \frac{\rho v^2}{2g} + 10\xi_f \frac{a}{d_e} \frac{\rho v^2}{2g}. \quad (23)$$

In the formula, ΔP_f is the friction resistance (Pa), ξ_f is the coefficient of friction resistance, α is the effective length of muffler(m), d_e (for a rectangular tube: $d_e = 2ab/(a+b)$) is the equivalent diameter of muffler passage (m), the air density is $\rho = 1.29\,\text{kg/m}^3$, v is the velocity of flow (m/s), and g is the acceleration of gravity.

Data about performance parameter of the screw air compressor before and after transformations are obtained through theoretical calculations.

Propagation of every performance propagation property for the screw air compressor after multiple active transformations is got according to the calculation steps discussed in the previous sections. Parameters set of multiple active transformations is got through doing multiple active transformations to the muffler discussed in Section 4

$$\begin{aligned} V_i\left(v_j, v_j'\right) &= \{V_1, V_2, \ldots, V_6\} = \left\{\left\{\left(v_1, v_1'\right), \ldots, \left(v_5, v_5'\right)\right\}; \ldots; \left\{\left(v_1, v_1'\right), \ldots, \left(v_8, v_8'\right)\right\}\right\} \\ &= \{\{(9500, 5000), (60, 85), (60, 85), (180, 340), (140, 161.5)\}; \\ &\quad \{(9500, 6000), (60, 85), (60, 85), (180, 340), (140, 161.5)\}; \\ &\quad \{(9500, 5000), (60, 85), (60, 85), (180, 340), (140, 161.5), (0, 80.75), (0, 40.35)\}; \\ &\quad \{(9500, 6000), (60, 85), (60, 85), (180, 340), (140, 161.5), (0, 80.75), (0, 40.35)\}; \\ &\quad \{(9500, 5000), (60, 85), (60, 85), (180, 340), (140, 161.5), (0, 80.75), (0, 40.35), (161.5, 58.5)\}; \\ &\quad \{(9500, 6000), (60, 85), (60, 85), (180, 340), (140, 161.5), (0, 80.75), (0, 40.35), (161.5, 58.5)\}\}. \end{aligned} \quad (24)$$

The weight of the active transformation is gained using the AHP method:

$$\begin{aligned} \text{AHP}\left(\sum_{i=1}^{h} \lambda_i\right) = \{&(0.030, 0.096, 0.074, 0.520, 0.280); (0.024, 0.109, 0.090, 0.427, 0.242, 0.059, 0.049); \\ &(0.018, 0.132, 0.114, 0.353, 0.195, 0.041, 0.036, 0.111)\}. \end{aligned} \quad (25)$$

Propagation degree of the performance attribute is calculated using formula (14).

The influence of active transformation on propagation properties can be measured by the degree of propagation. Active transformation will cause several propagation transformations because of the correlation between exhaust and the intake pressure. The degree of propagation is also a measure to measure the degree of the influence of several propagations taking place in a sequence.

Six transformation programs are generated by analyzing the most similar product LG-6.3/10 in which noise, intake pressure, weight, and exhaust are identified as the propagation properties. Then the six programs are classified using the method discussed in Section 4 by calculating their respective correlation functions.

Noise performance extension classification of the screw air compressor LG-6.3/10 caused by propagation transformation is as follows.

Negative quantitative:

$$\begin{aligned} E_{-}^{4}\left({}_{m_4}T_a\right) &= \left\{m_4 \mid m_4 \in M_O, y = k_4\left(m_4\right) \le 0, \right. \\ &\qquad \left. y' = k_j\left({}_{m_4}T_a m_4\right) < 0\right\}, \\ E_{-}^{4}\left({}_{m_4}T_b\right) &= \left\{m_4 \mid m_4 \in M_O, y = k_4\left(m_4\right) \le 0, \right. \\ &\qquad \left. y' = k_j\left({}_{m_4}T_b m_4\right) < 0\right\}, \\ E_{-}^{4}\left({}_{m_4}T_e\right) &= \left\{m_4 \mid m_4 \in M_O, y = k_4\left(m_4\right) \le 0, \right. \\ &\qquad \left. y' = k_j\left({}_{m_4}T_e m_4\right) < 0\right\}, \\ E_{-}^{4}\left({}_{m_4}T_f\right) &= \left\{m_4 \mid m_4 \in M_O, y = k_4\left(m_4\right) \le 0, \right. \\ &\qquad \left. y' = k_j\left({}_{m_4}T_f m_4\right) < 0\right\}. \end{aligned} \tag{26}$$

Positive qualitative:

$$\begin{aligned} \dot{E}_{+}^{4}\left({}_{m_4}T_c\right) &= \left\{m_4 \mid m_4 \in M_O, y = k_4\left(m_4\right) \le 0, \right. \\ &\qquad \left. y' = k_j\left({}_{m_4}T_c m_4\right) > 0\right\}, \\ \dot{E}_{+}^{4}\left({}_{m_4}T_d\right) &= \left\{m_4 \mid m_4 \in M_O, y = k_4\left(m_4\right) \le 0, \right. \\ &\qquad \left. y' = k_j\left({}_{m_4}T_d m_4\right) > 0\right\}. \end{aligned} \tag{27}$$

Weight performance extension classification of the screw air compressor LG-6.3/10 caused by propagation transformation is as follows.

Positive quantitative:

$$\begin{aligned} E_{+}^{5}\left({}_{m_5}T_a\right) &= \left\{m_5 \mid m_5 \in M_O, y = k_5\left(m_5\right) > 0, \right. \\ &\qquad \left. y' = k_5\left({}_{m_5}T_a m_5\right) > 0\right\}, \\ E_{+}^{5}\left({}_{m_5}T_c\right) &= \left\{m_5 \mid m_5 \in M_O, y = k_5\left(m_5\right) > 0, \right. \\ &\qquad \left. y' = k_5\left({}_{m_5}T_c m_5\right) > 0\right\}, \\ E_{+}^{5}\left({}_{m_5}T_e\right) &= \left\{m_5 \mid m_5 \in M_O, y = k_5\left(m_5\right) > 0, \right. \\ &\qquad \left. y' = k_5\left({}_{m_5}T_e m_5\right) > 0\right\}, \\ E_{+}^{5}\left({}_{m_5}T_f\right) &= \left\{m_5 \mid m_5 \in M_O, y = k_5\left(m_5\right) > 0, \right. \\ &\qquad \left. y' = k_5\left({}_{m_5}T_f m_5\right) > 0\right\}. \end{aligned} \tag{28}$$

Negative qualitative:

$$\begin{aligned} \dot{E}_{-}^{5}\left({}_{m_5}T_b\right) &= \left\{m_5 \mid m_5 \in M_O, y = k_5\left(m_5\right) \ge 0, \right. \\ &\qquad \left. y' = k_5\left({}_{m_5}T_b m_5\right) < 0\right\}, \\ \dot{E}_{-}^{5}\left({}_{m_5}T_d\right) &= \left\{m_5 \mid m_5 \in M_O, y = k_5\left(m_5\right) \ge 0, \right. \\ &\qquad \left. y' = k_5\left({}_{m_5}T_d m_5\right) < 0\right\}. \end{aligned} \tag{29}$$

Because the intake pressure is changed due to the structure change, exhaust is changed as well. Thus parameters about the exhaust of LG-6.3/10 also change due to the propagation transformation. This situation of having two propagations is termed quadratic propagation. The extension classification caused by multiple propagations is as follows.

Positive quantitative:

$$\begin{aligned} &E_{+}^{1}\left({}_{m_8}T_a \Longrightarrow {}_{m_1}T_{m_8}\right) \\ &\quad = \left\{m_1 \mid m_1 \in M_O, y = k_1\left(m_1\right) > 0, \right. \\ &\qquad \left. y' = k_1\left(\left({}_{m_8}T_a \Longrightarrow {}_{m_1}T_{m_8}\right) m_1\right) > 0\right\}, \\ &E_{+}^{1}\left({}_{m_8}T_b \Longrightarrow {}_{m_1}T_{m_8}\right) \\ &\quad = \left\{m_1 \mid m_1 \in M_O, y = k_1\left(m_1\right) > 0, \right. \\ &\qquad \left. y' = k_1\left(\left({}_{m_8}T_b \Longrightarrow {}_{m_1}T_{m_8}\right) m_1\right) > 0\right\}, \\ &E_{+}^{1}\left({}_{m_8}T_c \Longrightarrow {}_{m_1}T_{m_8}\right) \\ &\quad = \left\{m_1 \mid m_1 \in M_O, y = k_1\left(m_1\right) > 0, \right. \\ &\qquad \left. y' = k_1\left(\left({}_{m_8}T_c \Longrightarrow {}_{m_1}T_{m_8}\right) m_1\right) > 0\right\}, \\ &E_{+}^{1}\left({}_{m_8}T_e \Longrightarrow {}_{m_1}T_{m_8}\right) \\ &\quad = \left\{m_1 \mid m_1 \in M_O, y = k_1\left(m_1\right) > 0, \right. \\ &\qquad \left. y' = k_1\left(\left({}_{m_8}T_e \Longrightarrow {}_{m_1}T_{m_8}\right) m_1\right) > 0\right\}, \end{aligned}$$

$$E_{+}^{1}\left({}_{m_8}T_f \Longrightarrow {}_{m_1}T_{m_8}\right) = \left\{m_1 \mid m_1 \in M_O, y = k_1\left(m_1\right) > 0, \quad y' = k_1\left(\left({}_{m_8}T_f \Longrightarrow {}_{m_1}T_{m_8}\right) m_1\right) > 0\right\}. \tag{30}$$

Negative qualitative:

$$\dot{E}_{-}^{1}\left({}_{m_8}T_d \Longrightarrow {}_{m_1}T_{m_8}\right) = \left\{m_1 \mid m_1 \in M_O, y = k_1\left(m_1\right) \geq 0, \quad y' = k_1\left(\left({}_{m_8}T_d \Longrightarrow {}_{m_1}T_{m_8}\right) m_1\right) < 0\right\}. \tag{31}$$

Transformation programs are got by analyzing multiple active transformations and it is described as $(l_1^3 \wedge l_2 \wedge l_3^1 \wedge l_4^1 \wedge l_5^1 \wedge l_6^1 \wedge l_7^1 \wedge l_8^1 \wedge l'')$. Changes in the noise attribute are positive qualitative, changes in the weight attribute are positive quantitative, and changes in the exhaust attribute are positive quantitative. Propagation knowledge extracted from the active transformation process is described as follows:

$$\begin{aligned}&\left(c = T_4 \wedge T_1 \wedge T_1 \wedge T_1 \wedge T_1 \wedge T_2 \wedge T_2 \wedge T_1 \wedge T_3\right)\\ &\quad \Longrightarrow \left(\left({}_{m_8}T_c \Longrightarrow {}_{m_1}T_{m_8}\right) \wedge {}_{m_4}T_c \wedge {}_{m_5}T_c\right)\\ &\quad \Longrightarrow K(P) > 0.\end{aligned} \tag{32}$$

(4) *Simulation and Verification.* In this paper, Virtual.Lab Rev10 is used to verify if the intake expanded muffler, after changes and optimization on its material and structure, can meet customers' requirements on the noise performance.

(1) 3D picture of expanded muffler after changes and optimization is shown in Figure 4. The simplified discrete solid model of the expanded muffler computational domain after transformation and optimization is shown in Figure 5.

(2) The hypermesh is used to perform mesh division on the discrete solid model of the expanded muffler computational domain, and the resultant meshes are saved in the certain documents (as .bdf). These documents are imported into the Acoustics → AHB model of Virtual.Lab Rev10. The picture of the sound pressure cloud after calculating and transmission loss of muffler are shown in Figures 6 and 7.

Virtual.Lab Rev10 is used to simulate the transmission loss of the muffler after transformation to get the transmission loss curve at different frequencies. It is shown that the schemes after transformations meet the requirement of the low-frequency noise. It also complies with the trend of the extension configuration design elements discussed above.

7. Implementation of a Prototype System

A prototype system for the configuration design and conflict resolution for large screw air compressor has been developed based on a previous ET-based design system V1.0. The graphical user interface of the prototype system is shown in Figure 8, and graphical user interface for conflict resolution design system is shown in Figure 9.

The prototype system includes

(1) the module for degree-of-matching calculation between customers' performance requirements and the performance attribute values of a specific product: the graphical user interface in this module is used to explain the process as well as the operations involved in this module; the same idea is used for the explanation of other modules as well;

(2) the conflict resolution module which involves the transformations of performance attributes based on propagation analysis;

(3) the module for extracting knowledge from the propagation process;

(4) the module for outputting generated design schemes.

8. Conclusions and Future Work

The description of performance requirements and the resolution of conflicts caused by different requirements hold to key to effectively meeting new market needs. In this paper, a method for conflict resolution in the PDD is proposed based on the transformation of performance attributes and the propagation of changes. The degree of how a product meets a specific performance requirement is evaluated using a measure of similarity. To fit with the purpose of retrieving product cases from a case base, the traditional distance measure in the Extension Theory is improved in terms of both the left-side distance and the right-side distance representations. In addition, the correlation between product performance attributes and product structure parameters is established to identify the methods for performing transformations on the current design space and use change propagation to coordinate conflicts. On this basis, the method for coordinating conflicts based on the Extension Theory is devised and the method for mining propagation knowledge is developed. These methods have been evaluated in a screw air compressor design problem and implemented in a prototype system for conflict resolution for PDD.

It is shown in the evaluation that the improved distance representation achieves improved retrieval precision. The proposed conflict resolution method is feasible and implementable. As shown in the case study, the method effectively performs several transformations during the coordination of conflict for the screw air compressor and utilizes the propagation of design changes to resolve conflicts and thus improve overall performance. It also shows that the Extension Theory provides effective formal model for the resolving of conflict problems especially in the design domain where problems are highly coupled and multidimensional. This work is still at an early stage and a lot of work needs to be done in the future. Firstly, we will improve the case base and include more performance attributes in each case. Secondly, more complicated design problems will be analyzed and the

proposed method will be used in these problems. Thirdly, the functionality of the prototype system will be improved.

Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

Acknowledgment

The paper is sponsored by the National Science Foundation Project of China (no. 51275477).

References

[1] W. Anan, "Value engineering," *Engineering Management Joumal*, vol. 3, no. 4, pp. 171–175, 2003.

[2] Y. Xie, "Study on the modern design theory and methodology," *Chinese Journal of Mechanical Engineering*, vol. 40, no. 4, pp. 1–9, 2004.

[3] SuhNP, *The Prinei Plesofdesign*, Oxford University Press, New York, NY, USA, 1990.

[4] Z. Wei, J. Tan, and Y. Feng, "Method of conceptual design of mechanic product driven by generalized performance," *Chinese Journal of Mechanical Engineering*, vol. 44, no. 5, pp. 1–8, 2008.

[5] Y. Xie, "Product performance features and modern design," *China Mechanical Engineering*, vol. 11, no. 1-2, pp. 26–32, 2000.

[6] X. Meng, Y. Xie, and X. Dai, "Methodology of designing for time-varying performance of complex products," *Chinese Journal of Mechanical Engineering*, vol. 46, no. 1, pp. 128–133, 2010.

[7] S.-L. Yang and C. Fu, "Constructing confidence belief functions from one expert," *Expert Systems with Applications*, vol. 36, no. 4, pp. 8537–8548, 2009.

[8] J. Li and Y. Zhou, "The evolution of rules for conflicts resolution in self-organizing teams," *Expert Systems with Applications*, vol. 39, no. 1, pp. 239–246, 2012.

[9] G. W. Tan, C. C. Hayes, and M. Shaw, "An intelligent-agent framework for concurrent product design and planning," *IEEE Transactions on Engineering Management*, vol. 43, no. 3, pp. 297–306, 1996.

[10] S. Cooper and A. Taleb-Bendiab, "CONCENSUS: multi-party negotiation support for conflict resolution in concurrent engineering design," *Journal of Intelligent Manufacturing*, vol. 9, no. 2, pp. 155–159, 1998.

[11] S. T. C. Wong, "Coping with conflict in cooperative knowledge-based systems," *IEEE Transactions on Systems, Man, and Cybernetics A*, vol. 27, no. 1, pp. 57–72, 1997.

[12] G.-L. Xiong and H.-B. Ma, "Research on conflict negotiation in concurrent design," *Control and Decision*, vol. 16, pp. 721–724, 2001.

[13] M. Klein, "Conflict management as part of an integrated exception handling approach," *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 9, no. 4, pp. 259–267, 1995.

[14] R.-H. Tan and F. Liu, "Ideas generation by integrating needs evolution with functions evolution," *Computer Integrated Manufacturing Systems*, vol. 17, no. 10, pp. 2093–2100, 2011.

[15] G. Ahshuller, *And Suddenly the Inventor Appeared-TRIZ, The Theory of Inventive Problem Solving*, Technical Innovation Center, Worcester, Mass, USA, 1996.

[16] J. Terninko, A. Zusman, and B. Zlotin, *Systematic Innovation—An Introduction to TRIZ*, St Lucie Press, New York, NY, USA, 1998.

[17] Y. Salamatov, *TRIZ: The Right Solution at the Right Time: A Guide to innovative Problem Solving*, Insytec BV, Amsterdam, The Netherlands, 1999.

[18] S. D. Savransky, *Engineering of Creativity*, CRC Press, Boca Raton, Fla, USA, 2000.

[19] J. Hesselbach and C. Hermann, "Enhancement in industrial PSS design based on TRIZ: a case study," in *Proceedings of the 3rd CIRP International Conference on Industrial Product Service Systems*, pp. 225–230, Springer, 2011.

[20] H.-T. Chang and J. L. Chen, "The conflict-problem-solving CAD software integrating TRIZ into eco-innovation," *Advances in Engineering Software*, vol. 35, no. 8-9, pp. 553–566, 2004.

[21] W. Cai, C. Yang, and W. Lin, *Extension Engineering Method*, Science Press, Beijing, China, 1997.

[22] H.-T. Chang and J. L. Chen, "An approach combining extension method with TRIZ for innovative product design," *Journal of the Chinese Society of Mechanical Engineers*, vol. 25, no. 1, pp. 13–22, 2004.

[23] Y. Zhao and S. Nan, *Extension Design*, Science Press, Beijing, China, 2010.

[24] J. Chen, Y. Zhao, F. Li, and J. Li, "Transforming bridge-based conflict resolution for product green design," *Journal of Mechanical Engineering*, vol. 46, no. 9, pp. 132–142, 2010.

[25] Y. Zhao, N. Su, F. Zhang, and J. Chen, "Configuration design method for product family based on extension case reasoning," *Journal of Mechanical Engineering*, vol. 46, no. 15, pp. 146–154, 2010.

[26] Y. Chunyan and C. Wen, *Extenics Theory, Method and Application*, Science Press, Beijing, China, 2013.

[27] H. Wang, *Research and Application of Performance Conduction Configuration Design Based on Extension Knowledge Mining*, ZheJiang University of Technology, Hangzhou, China, 2013.

[28] D. Ma, *The Engineering Manual of Noise and Vibration Control*, Machine Press, Beijing, China, 2002.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2014, Article ID 587263, 6 pages
http://dx.doi.org/10.1155/2014/587263

Research Article

Advanced Nonlinear Dynamic Analysis of Arch Dams considering Joints Effects

Mohammad Amin Hesari,[1] Mohsen Ghaemian,[2] and Abolfazl Shamsai[1]

[1] *Department of Technical and Engineering, Science and Research Branch, Islamic Azad University, Tehran 51547-94889, Iran*
[2] *Department of Civil Engineering, Sharif University of Technology, Tehran 51547-94889, Iran*

Correspondence should be addressed to Mohammad Amin Hesari; amin.hesari@gmail.com

Received 24 August 2013; Accepted 17 November 2013; Published 12 January 2014

Academic Editor: Siamak Talatahari

Influence of joints behavior on arch dams operation during the earthquakes is investigated. The case study is the Karun-1 double curvature arch dam with the height of 200 meters. The arch dam-foundation-reservoir systems are modeled with and without joints and estimate the effects of contraction and lift joints on stresses and displacements response histories for assessing the earthquake performance. According to nolinear dynamical analysis results, inclusion of the contraction and lift joints considerably influenced the dam response.

1. Introduction

The finite element method (FEM) is a numerical method that can be used to solve different kinds of engineering problems in the stable, transient, linear, and nonlinear cases [1]. Among finite element method software programs, ABAQUS is known as one of the most precise and practicable software programs in industry and academic researches. It is used for high capability to advance nonlinear dynamic analysis such as earthquake and water wave loading on structures [2].

An arch dam is a solid concrete dam, curved upstream in plan. In addition to resisting part of the pressure of the reservoir by its own weight, it obtains a large measure of stability by transmitting the remainder of the water pressure and dynamic loads by arch action into the canyon walls. The complete necessity of high safety, economical design, complex of designing, and its application increases the importance of arch dams. Successful arch action is dependent on a unified monolithic structure and a special care must be taken in the construction of an arch dam to ensure that no structural discontinuities exist, such as open joints or cracks. By noting that one of the important parameters in dam construction projects is the earthquake, therefore, to protect against human loss or economic damages, it is necessary to have a dynamical analysis for this massive structure. Providing the safety of arc against earthquake has been noticed by many researchers and engineers. National information service for earthquake engineering (NISEE) in Berkley university is one of the most reliable centers that has a lot of scientific reports about the discussed subject. It can be noted that dynamic reservoir interaction with Monticello dam was studied by Clough et al. [3] and Study of joint opening effects on earthquake response of arch dams was done by Fenves et al. [4]. Sheng et al. [5] studied the effects of contraction joints on earthquake response of the arch dams. Ĺger and Seydou [6] presented the hybrid dam displacement model to simulate the seasonal thermal displacements of gravity concrete dams. They used a model for predicting the behavior of dam under extreme thermal effects which cannot happen in real conditions. Other researches conducted in this area can be referred to Akkose et al. [7] and by Wang et al. [8] and studied damage assessment of a concrete arch dam through nonlinear incremental dynamic analysis by Ghaemian and Alembagheri [9]. Chopra [10] studied earthquake analysis of arch dams with factors considered.

In this research Karun-1 arch dam is modeled in two cases by using ABAQUS package. First dam without joints

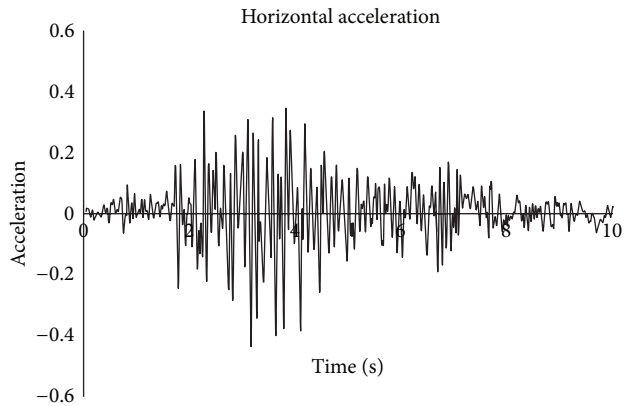


Figure 1: Horizontal acceleration time histories of earthquake.

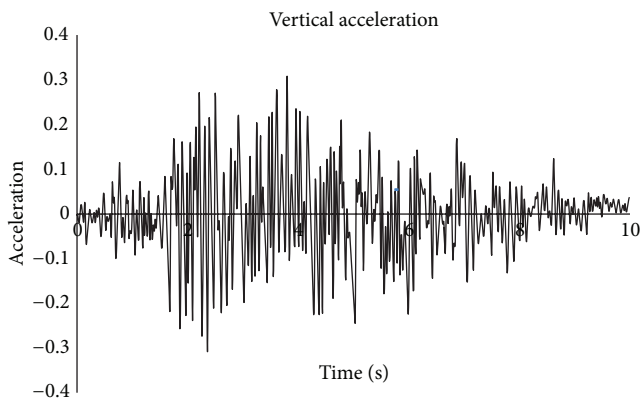


Figure 2: Vertical acceleration time histories of earthquake.

and second dam body with joints. For the prevention of initiation and growth of thermal cracks in dam body, it has been normal procedure to provide vertical contraction joints in arch dams at approximately 15 to 20 m spacing. In this way, continuous dam body is separated to several vertical blocks mainly named monoliths in scientific literatures. Thermal analysis has been done to study the effect of contraction joints on the dam safety, then dynamic and thermal loads are applied to 3D exact simulation of the geometric model of the dam. Finally the effect of joints on responses of dam against earthquake is compared and investigated.

2. Acceleration Time-Histories of Earthquake

Applied records on dam are the horizontal and vertical components of the Koyina earthquake. The acceleration of this earthquake was 0.5 g and it should be classified as a design basis earthquake (DBE) with happening probability of 20% and 200 years return period. Vertical and horizontal acceleration of the earthquake have been given in Figures 1 and 2.

3. Modeling of Dam by ABAQUS Software

For modeling we need a proper case study to have a real simulated model. Karun-1 dam is created by noting its design sections. It is double curvature arch dam with 200 m height. Additional information is represented in Table 1 (see Figures 3 and 4).

Table 1: Geometrical and material specification of Karun-1 dam.

Dam type	Double curvature arch dam	Thickness at crest	6 m
Length of crest	372 m	Normal water level	177.5 m
Height	200 m	Concrete Poisson's ratio	0.2
Thickness at base	33.5 m	Concrete density	2400 Kg/m^3

Figure 3: View of Karun-1 dam.

To have compatibility between the geometry of the dam and the model, 8-node, 3-dimensional elements (C3D8RH) are used in ABAQUS.

4. Modeling of Abutments

Abutments modeling is done by considering the ratio of elasticity modulus of bed rock over elasticity modulus of dam body concrete (E_r/E_c). The larger the ratio, the stiffer the bedrock, and it will result in low height for modeling of the abutments. It means that, with less bed rock stiffness, to have precise interaction between foundation and dam, we need more height to model the abutments. In this study the ratio is between 1 and 2. So, the dimensions of the abutments for modeling in any direction are considered equal to the dam height (U.S. Army, 1994).

Damping coefficient is calculated by Riley method. Damping coefficient can be obtained by means of linear combination of mass matrix and stiffness matrix. Coefficients of these linear components are calculated from frequency modes of dam:

$$C = \alpha M + \beta K. \quad (1)$$

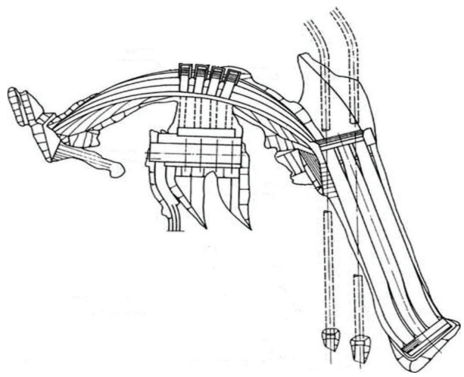


Figure 4: Plan of Karun-1 dam.

Table 2: Five-mode frequency of dam.

Frequency analysis	Mode	1	2	3	4	5
	Frequency	1.76799	2.14547	2.986	3.3423	3.7177

α and β are Riley linear combining coefficients and calculated from (2) and (3):

$$\alpha = 2\xi\omega_i \cdot \frac{\omega_j}{\omega_i + \omega_j}, \tag{2}$$

$$\beta = \frac{2\xi}{\omega_i + \omega_j}. \tag{3}$$

In these equations ξ is the damping ratio and in dynamical analysis is assumed to be 5%. ω_i and ω_j are two angular frequencies of vibration modes. First five frequency modes that we extracted from model are represented in Table 2.

In Karun-1 dam dynamic analysis Riley damping coefficients are calculated from first and third dam vibrating modes and critical damping ratio which was 5%: $\alpha = 0.1110$ and $\beta = 0.0210$ [11].

5. Nonlinear Joint Model

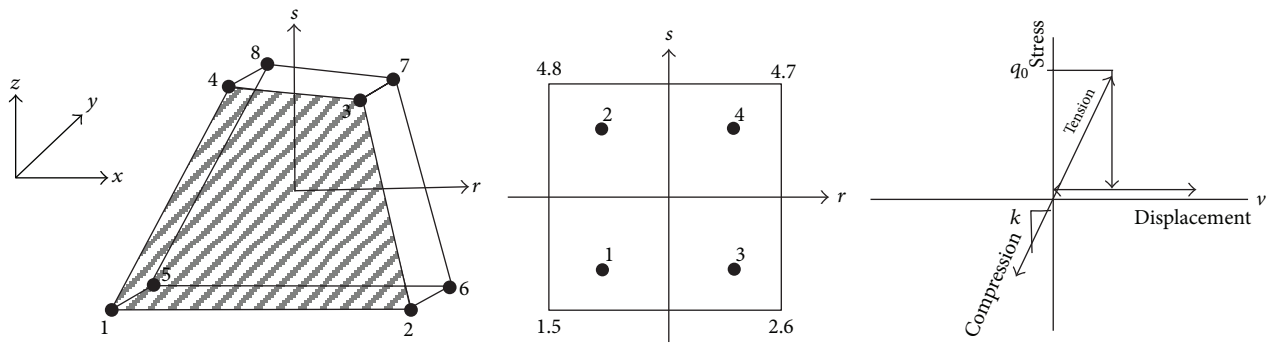
Concrete arch dams are constructed as individual cantilever blocks separated by vertical contraction joints. The contraction joints usually are grouted at the completion of the dam or in stages and may also include shear keys for additional resistance. However, contraction joints have limited tensile resistance and can open and close repeatedly during the earthquake ground shaking that produces net tensile forces across the joints. Such opening and closing of the joints are transient in nature but if severe can lead to unstable cantilever blocks and concrete crushing due to impact. The joint opening relieves tensile stresses across the joint but compressive stresses are increased due to the impact and reduced contact surfaces. This nonlinear behavior therefore needs to be investigated to ensure that the amount of joint opening and compressive stresses are not excessive. The contraction joint opening is modeled using a nonlinear joint element shown in Figure 5 [4]. The nonlinear joint element consists of two coincident surfaces, each defined by four nodes that may lie on a plane (See Figure 5(a)). The element stiffness properties, displacements, and stresses are computed at four Gauss integration points shown in Figure 5(b). The relative displacements between the two element surfaces (v) and produce stresses (q), according to the nonlinear relationship described in Figure 5(c). As shown in this figure, the element is characterized by the tensile strength, q_0, and the stiffness, k. The tensile strength q_0 can be selected to represent different joint types. For example, $q_0 = 0$ represents an ungrouped joint or a grouted joint with zero tensile strength, whereas a nonzero q_0 represents a grouted joint for which some tensile strength can be assumed. Finally, a large value of q_0 may be used to simulate a linear analysis for which the joints are not permitted to open. The nonlinear joint elements are placed between cantilever blocks to model opening and closing of joints.

The dam model is developed using mesh generation capability of the ABAQUS program. It includes three layers of 8-node solid elements through the dam thickness (C3D8RH). The finite-element mesh of the dam is arranged along horizontal and vertical planes to model the joints. The vertical planes are oriented in the radial direction same as the contraction joints. This way, when needed, nonlinear joint elements can easily be inserted along the horizontal lift lines and the vertical contraction joints. Because of space limitations of paper, three studied points of dam are node 49 in up stream of dam at midpoint of crest level, node 43 on right side of crest level, and node 29 at midpoint of dam body (see Figure 6).

6. Computation of Nonlinear Earthquake Response

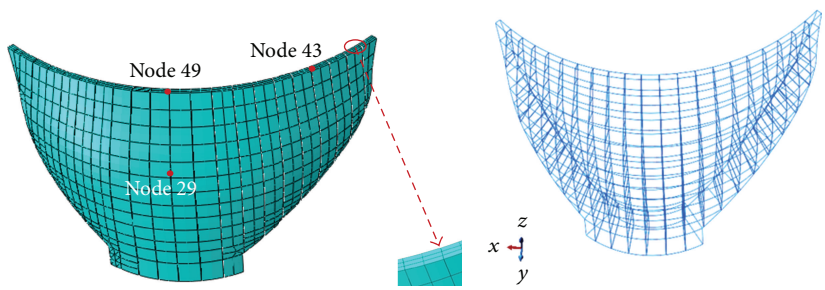
6.1. Dynamic Analysis and Extraction of Stress Time History. According to International Committee on Large Dams (ICOLD) for choosing the earthquake parameters, dam should withstand against applied credible earthquake. Some damages are allowed in limited areas for maximum design earthquake. These damages should be in a way that dam could keep impounded water. The analysis began with the applied of gravity and hydrostatic load and temperature loads as the initial conditions, followed by the step-by-step nonlinear time-history dynamic analysis for the dam with and without joints. Figure 7 shows envelope of maximum principal stress without joints and Figure 8 shows envelope of maximum principal stress with contraction and lift joints under dynamic analysis.

The figures show that stresses are reduced considerable in the model with joints. For both cases the maximum stress is near the abutments and minimum is at the midpoints of crest that are significant inclusion of joints. History graphs of stress at the studied points are represented in Figures 9, 10, and 11. The figures show that the effects of the joints are obvious, as stresses are considerably reduced in the model with joints for all of the studied points and they are almost zero. But in the model without joints the stresses are increased with high vibrations.



(a) Local and global coordinate systems (b) Integration points (c) Stress-relative displacement relationship

Figure 5: Nonlinear joint elements.



(a) (b)

Figure 6: Finite-element meshes of dam body and studied points.

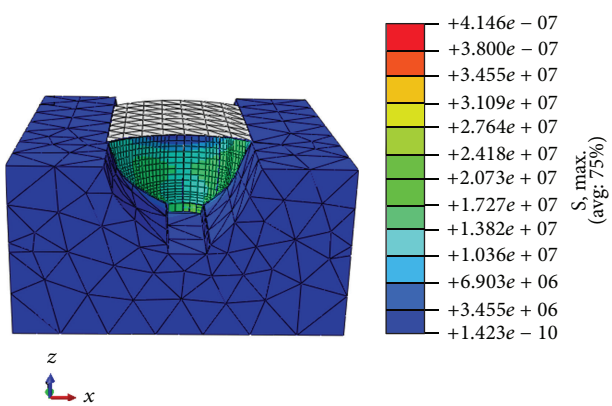


Figure 7: Envelope of maximum stress without joints.

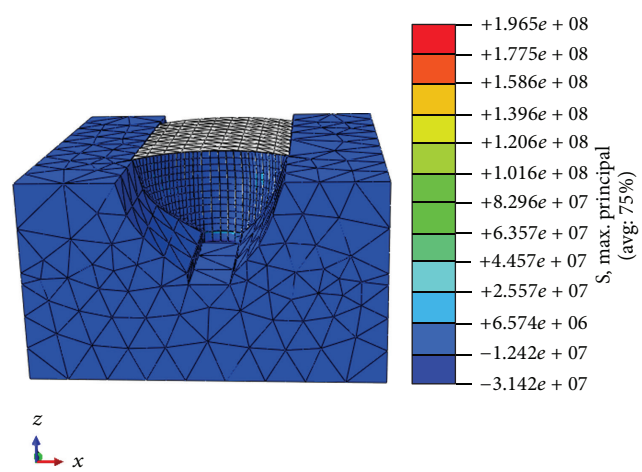


Figure 8: Envelope of maximum stress with joints.

6.2. Dynamic Analysis and Extraction of Displacement Time History. Results of dynamic analysis of the dam with and without joints are illustrated in Figures 12 and 13. In both models with and without joints, results show that displacement around abutment is near zero but maximum displacements happen in crest level especially at midpoint of crest. We see that displacements are distributed over the dam with maximum around crest. Displacements histories in reservoir direction (U2) at studied points are shown in Figures 14, 15, and 16. According to history graphs at node 43 in the dam with joints, displacements are increased with high vibration at midpoint of the dam (see Figure 16). The displacement history in the reservoir direction at node 29 and midpoint of crest (node 49) inclusion of joints is increased too (see Figures 14 and 15).

7. Conclusions

(1) According to the results of dynamic analysis and applied earthquake records, for both model of dam without joints and with contract and lift joints,

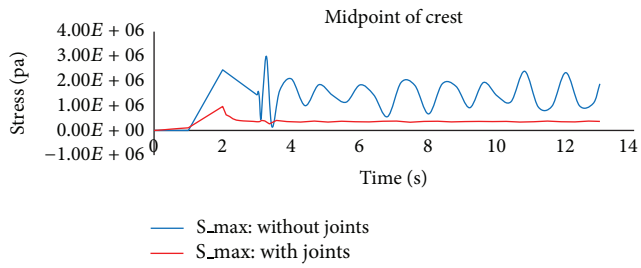


FIGURE 9: Stress history at midpoint of crest level (node 49) with and without joints.

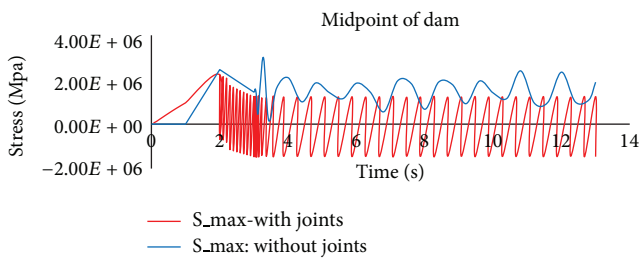


FIGURE 10: Stress history at midpoint of dam body (node 29) with and without joints.

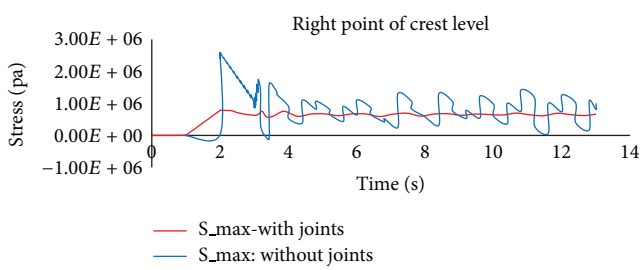


FIGURE 11: Stress history at right side of crest level (node 43) with and without joints.

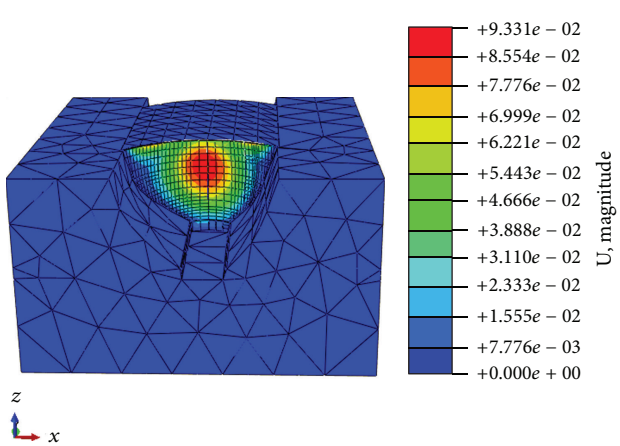


FIGURE 12: Envelope of displacement without joints.

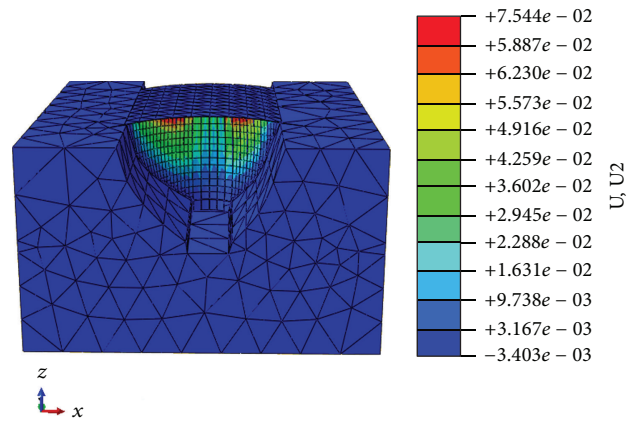


FIGURE 13: Envelope of displacement with joints.

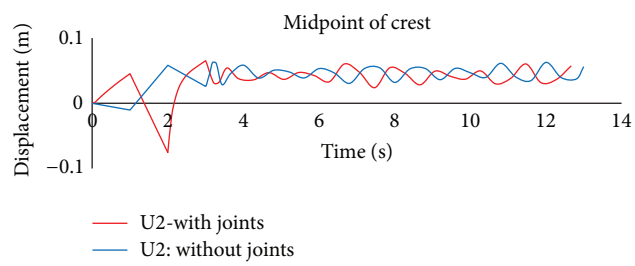


FIGURE 14: Displacement history at midpoint of crest (node 49) with and without joints.

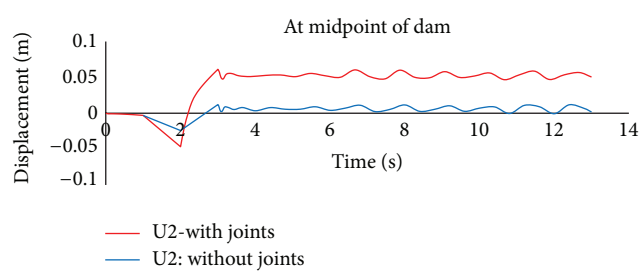


FIGURE 15: Displacement history at midpoint of dam body (node 29) with and without joints.

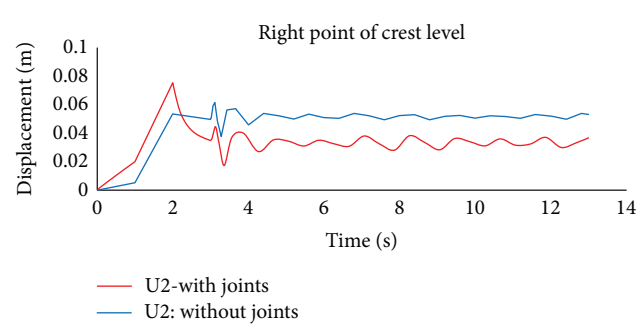


FIGURE 16: Displacement history at right side point of crest level (node 43) with and without joints.

maximum stresses occur around the abutments and maximum displacements are around the crest level especially at midpoint of the crest.

(2) With inclusion of the contraction and lift joints, tensile stresses will be considerably reduced. Allowing joints slippage should further reduce tensile stresses, if such capability is available.

(3) Inclusion of the contraction and lift joints will increase time history displacements especially at the right side point of the dam.

(4) Identification of and stability analyses for isolated blocks that may get loose and overtop as a result of excessive joint opening and lift line cracking are necessary for assessing the overall safety of the dam and its ability to retain the impounded water.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] Bathe, *Klaus—Jurgen*, Finite Element Procedures, Prentice Hall Inc, 1996.

[2] ABAQUS Analysis User's Manual, 2012.

[3] R. W. Clough, Y. Ghannat, and X.-F. Qui, *Dynamic Reservoir Interaction With Monticello Dam*, Report No. UCB/EERC-87/21, Earthquake Engineering Research Center, University of California at Berkley, 1987.

[4] G. L. Fenves, S. Mojtahedi, and R. B. Relmer, "Effect of contraction joints on earthquake response of an arch dam," *Journal of Structural Engineering*, vol. 118, no. 4, pp. 1039–1055, 1992.

[5] Z. Sheng, Y. Xu, and H. Liu, "Research on seismic failure pattern of arch dam with contraction joints by shaking table tests and numerical analysis," *Journal of Hydroelectric Engineering*, vol. 28, no. 3, pp. 69–74, 2009.

[6] P. Ĺger and S. Seydou, "Seasonal thermal displacements of gravity dams located in northern regions," *Journal of Performance of Constructed Facilities*, vol. 23, no. 3, pp. 166–174, 2009.

[7] M. Akkose, A. Bayraktar, and A. A. Dumanoglu, "Reservoir water level effects on nonlinear dynamic response of arch dams," *Journal of Fluids and Structures*, vol. 24, no. 3, pp. 418–435, 2008.

[8] J.-T. Wang, D.-D. Lv, F. Jin, and C.-H. Zhang, "Earthquake damage analysis of arch dams considering dam-water-foundation interaction," *Soil Dynamics and Earthquake Engineering*, vol. 49, pp. 64–74, 2013.

[9] M. Ghaemian and M. Alembagheri, "Damage assessment of a concrete arch dam through nonlinear incremental dynamic analysis," *Journal of Soil Dynamics and Earthquake Engineering*, vol. 44, pp. 127–137, 2012.

[10] A. K. Chopra, "Earthquake analysis of arch dams: factors to be considered," *Journal of Structural Engineering*, vol. 138, no. 2, pp. 205–214, 2012.

[11] J. A. Inaudi, E. E. Matheu, R. L. Poeppelman, and A. Matusevich, *Foundation Flexibility Effects on the Seismic Response of Concrete Gravity Dams*, U.S. Army Engineer Research and Development Center, Sacramento, Calif, USA, 2012.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 169823, 13 pages
http://dx.doi.org/10.1155/2013/169823

Research Article

Optimum Design of Geometrically Nonlinear Double-Layer Domes by Firefly Metaheuristic Algorithm

R. Kamyab Moghadas, A. Garakani, and M. Kalantarzadeh

Iranian Academic Center for Education, Culture and Research, Kerman 7616914111, Iran

Correspondence should be addressed to R. Kamyab Moghadas; r_kamyab_m@yahoo.com

Received 8 August 2013; Accepted 26 September 2013

Academic Editor: Siamak Talatahari

Size optimization of double-layer scallop domes subjected to static loading is focused in the present paper. As this type of space structures possesses a large number of the structural elements, optimum design of such structures results in efficient structural configurations. In this paper, optimization task is achieved using firefly algorithm (FA) by taking into account linear and nonlinear responses of the structure. In the nonlinear optimization process only the geometrical nonlinearity effects are included. The numerical results demonstrate that nonlinear optimization provides more efficient structures compared with the linear one.

1. Introduction

As the space structures are employed to cover wide span column-free areas, they have a huge number of structural elements, and, therefore, sufficient attention must be paid to systematic designing of these structures. For this purpose, design of space structures can be conveniently achieved by employing optimization techniques. It is obvious that an optimal design has a great influence on the economy and safety of all types of the structures. In this case, optimizing space structures results in more efficient structural configurations. The present study is devoted to design optimization of a specific type of space structures denoted as scallop domes [1]. Configuration of scallop domes includes alternate ridged and grooves that radiate from the centre. There are many actual examples of scallop domes that are constructed throughout the world.

In the recent years, much progress has been made in optimum design of space structures by considering linear behavior [2–4]. It is observed that some trusses show nonlinear behavior even in usual range of loading [5, 6]. Therefore, neglecting nonlinear effects in design optimization of these structures may lead to uneconomic designs.

In this study, double-layer scallop domes are designed for optimal weight considering linear and nonlinear behaviors. In the case of nonlinear optimization geometrical nonlinearity effects are taken into account. All of the structural optimization problems have two main phases: analysis and optimization. In the analysis phase, ANSYS [7] is employed and in the optimization phase, firefly algorithm (FA) [8] as an element of artificial intelligence (AI) is utilized. In the field of civil and structural engineering many applications of AI have been reported in the literature [9–15]. The FA is coded in MATLAB [16]. In this paper, the design variables are cross-sectional areas of the structural elements. The design constraints involved here are nodal displacements and element stresses constraints. Three illustrative examples are presented and the numerical results reveal that the nonlinear optimization of scallop domes results in efficient structural configuration compared with the linear optimization process.

2. Scallop Domes

The dome shown in Figure 1(a) illustrates a perspective view of a [diamatic] dome. The plan of this dome has been shown in Figure 1(b). As shown in Figure 1(c), the border members of this dome are situated along meridional ribs and circumferential rings. Meridional ribs and circumferential rings on a spherical surface of Figure 1(c) correspond to the meridians and parallels on the surface of the earth.

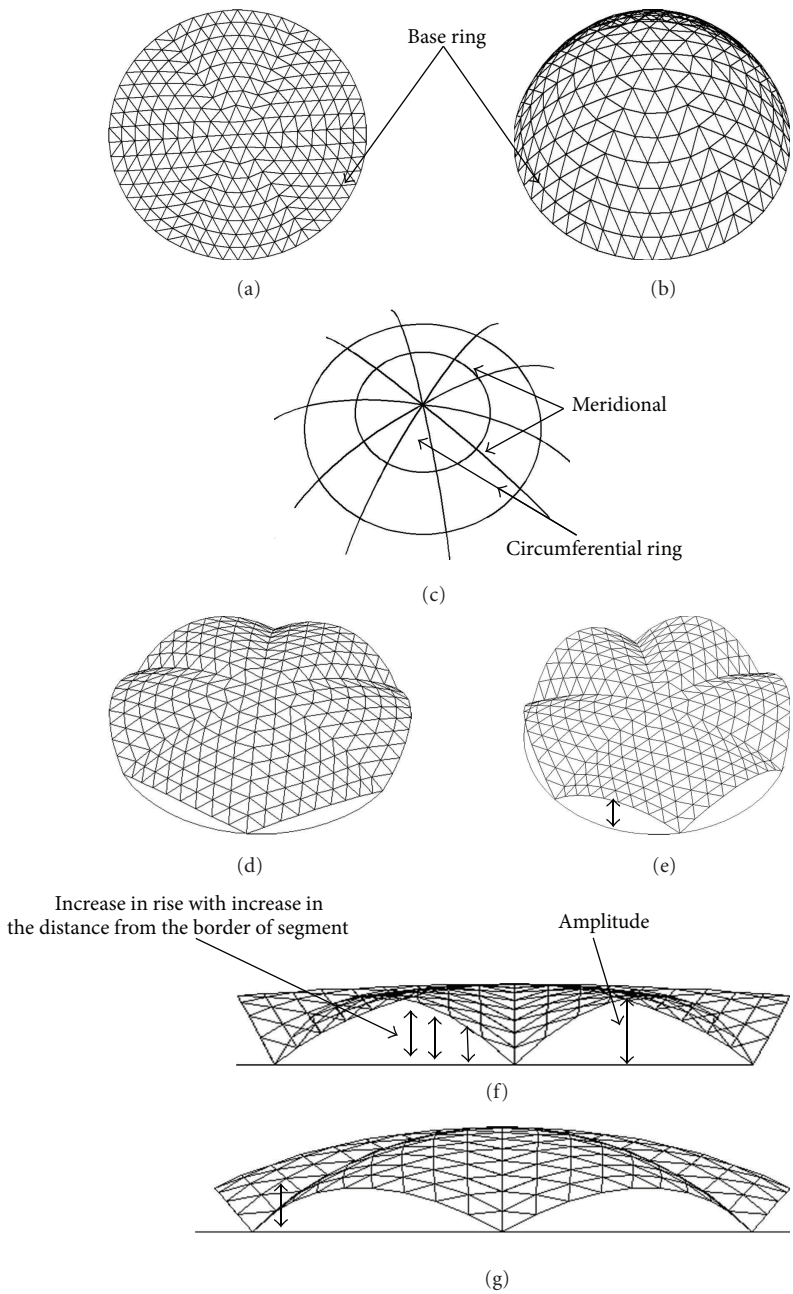


Figure 1: Example of scallop domes.

The perspective view of the dome configuration in Figure 1(d) is identical to the dome in Figure 1(a) in terms of the plan, the location of the meridional border members, and the height of the crown of the dome. However, as it is shown in Figure 1(d), the nodal points along every circumferential ring in the base diamatic dome are raised vertically such that the part of the ring between the borders is turned into an arch [1].

The dome that resulted from the arching of the members between the borders is called "scallop dome." In fact, the arching of the nodes happens in such a way that firstly, the nodal points on the segmental borders remain in their original position (in particular, the position of the crown of the dome remains unchanged) and, secondly, the rise of the arches increases with distance from the crown of the dome. Of course, as it is also shown in Figure 1(e), the amount of rise increases closing to the middle of the segment.

The maximum rise for the circumferential ring, which occurs at the middle of each segment, is referred to as the "the amplitude of the ring." The ring that is the furthest away from the crown, namely, the "base ring," has the largest amplitude. This amplitude is referred to as "the amplitude of the dome" (Figure 1(d)). In this figure, the underneath curve shows the initial position of the base ring before the arching of the segments.

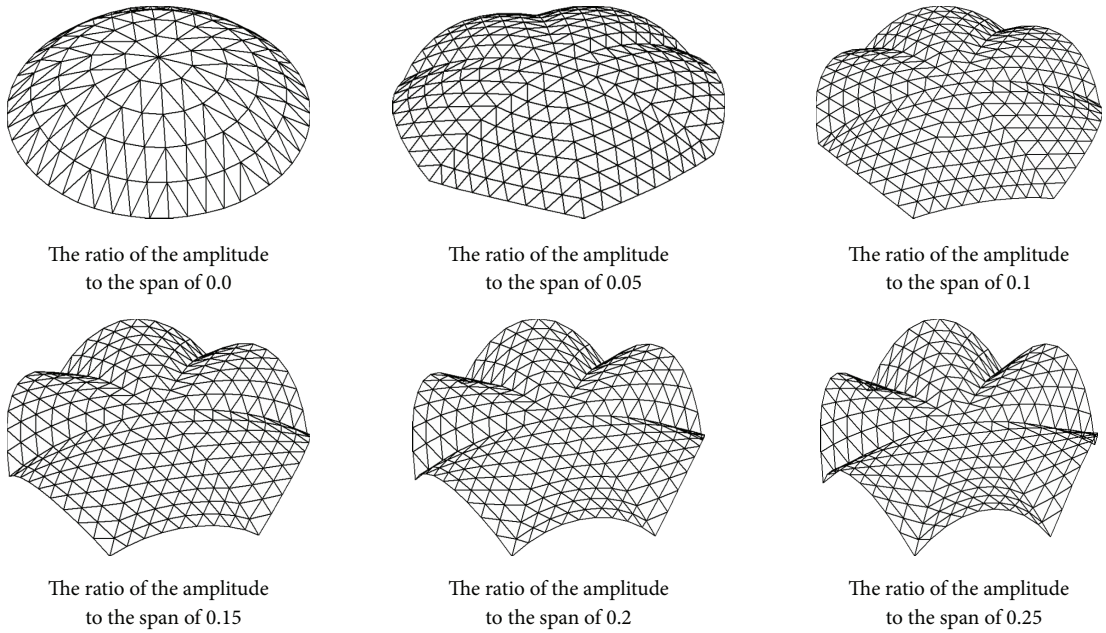


Figure 2: The variation of amplitude in scallop domes.

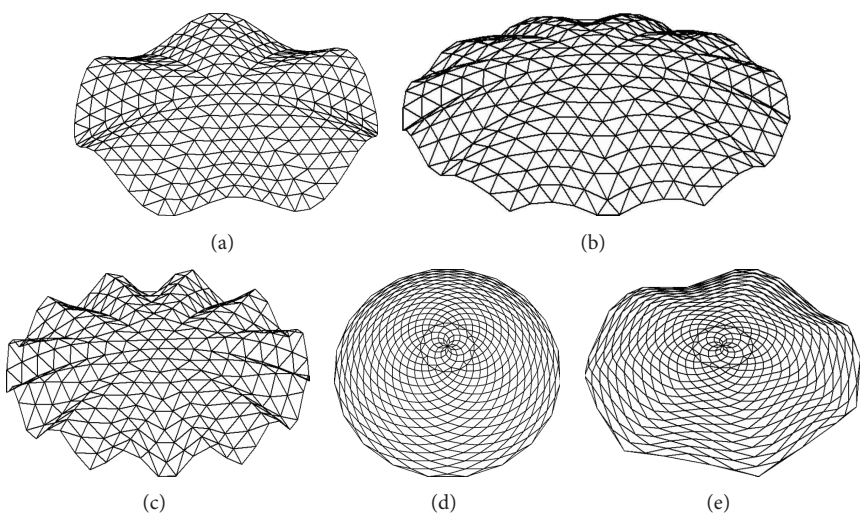


Figure 3: Arching styles of scallop domes.

The amount of amplitude considering the architectural or structural requirements could be different. For example, the scallop dome in Figure 1(f) is similar to the scallop dome in Figure 1(d) except for its amplitude which is reduced into half.

In Figure 2, some domes are shown with gradual increase in amplitude.

3. Arching Styles of Segments

The style of the variation of the height along with the circumferential ring can be accomplished in different ways. These height variations can be parabolic or sinusoidal. The scallop dome in Figure 3(a) is an example of parabolic variations [1].

The variation of height along the circumferential ring of the segment of the dome in Figure 1(d) is parabolic, whereas this variation of height in Figure 3(a) is sinusoidal.

The dome in Figure 3(c) like the one in Figure 3(b) includes 12 segments and the original domes of both are identical, but with different arching styles. Another example of sinusoidal scalloping is represented in Figure 3(e) which is obtained from scalloping the dome in Figure 3(c). Figure 4 clearly represents parabolic and sinusoidal arching. The horizontal base line in these Figs is along with a circumferential ring between the border points, which may be assumed to be obtained by straightening out the ring between the border points along the ring illustrates parabolic and sinusoidal curves in Figure 4(a) and Figure 4(b), respectively.

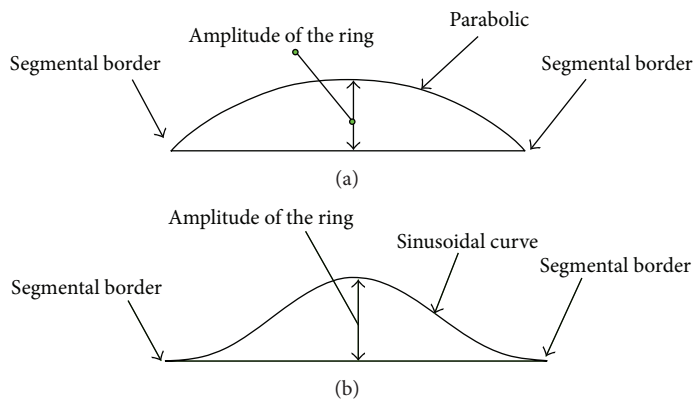


Figure 4: Parabolic and sinusoidal style of arching.

4. Theoretical Background of Nonlinear Analysis

In a linear static analysis it is assumed that the deflections and strains are very small and the stresses are smaller than the material yield stresses. Consequently, the stiffness can be considered to be independent of the displacements and forces and the finite element equilibrium equations are linear:

$$P = \mathbf{K}\delta, \tag{1}$$

where P, $\mathbf{K}$, and δ are the external load vector, stiffness matrix, and nodal displacements vector, respectively.

This linearity implies that any increase or decrease in the load will produce proportional increase or decrease in displacements, strains, and stresses. But it is clear that, in many structures, at or near failure (ultimate) loads, the deflections and the stresses do not change proportionately with the loads. Either the stresses are so high that they no longer obey Hooke's law or else there are such large deflections that the compatibility equations cease to be linear. These two conditions are called material nonlinearity and geometric nonlinearity, respectively. In this study, a finite elements model based on geometrical analysis of scallop domes is performed by ANSYS. In the nonlinear structural analysis process, instead of the linear strain, a nonlinear strain [17] is used. Since the strains are nonlinear functions of the displacements the stress to strain relationship is nonlinear. In this case, the stiffness is dependent on the displacements and the strains. Obviously, the solution of the displacements cannot be obtained in a single step. Instead, the analysis is carried out by the incremental method combined with some iterative equilibrium corrections at every step [17]. In this work, the Newton-Raphson method of solution is used. The steps of the solution procedure are as follows.

(1) Form tangent stiffness matrix ($\mathbf{K}_t$) with the latest values of displacements and stresses:

$$\mathbf{K}_t = \sum_{1}^{N_e} \int_e \mathbf{B}^T \mathbf{D} \mathbf{B}\, dv, \tag{2}$$

where $\mathbf{D}$ is material stiffness matrix.

(2) The incremental displacements equation is solved:

$$\Delta\delta = \mathbf{K}_t^{-1} \Delta P + \psi, \tag{3}$$

where ΔP is part of the load vector to be applied at the current increment and ψ is the residual forces.

(3) The incremental displacements are added to the total displacements:

$$\delta \longrightarrow \delta + \Delta\delta. \tag{4}$$

(4) The nonlinear strains are computed based on the latest estimate of the incremental strains $\Delta\varepsilon$.

(5) Total stresses are computed using the linear elastic stress strain relation:

$$\sigma \longrightarrow \sigma + \mathbf{D}\Delta\varepsilon. \tag{5}$$

(6) Contributions of the current Gauss point to the element internal forces are computed:

$$f_i = \int_e \mathbf{B}^T \sigma\, dv. \tag{6}$$

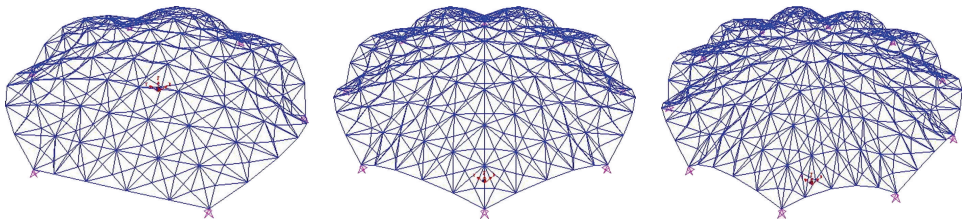
(7) Residual forces are computed:

$$\psi = \sum_{i=1}^{n_e} \left(f_i - F\right), \tag{7}$$

where F contains the external forces and n_e is the number of elements.

(8) Incremental loads are applied at the next increment and steps (1) to (7) are repeated until the process converges.

To increase the speed and accuracy of the nonlinear analysis, the applied loads are segmented into some loads termed substeps and then, in each substep, Newton-Raphson method is used.



(a) (b) (c)

Figure 5: Double-layer scallop dome with (a) 6, (b) 8, and (c) 10 segments.

Table 1: The available list of standard pipe profiles.

No.	Profile	A (cm^2)	r (cm)
1	TUBO-D33.70x2.6	2.540	1.1000
2	TUBO-D48.30x2.6	3.730	1.6200
3	TUBO-D60.30x3.2	5.740	2.0200
4	TUBO-D76.10x3.2	7.329	2.5799
5	TUBO-D82.50x3.2	7.972	2.8060
6	TUBO-D88.90x3.2	8.616	3.0321
7	TUBO-D101.6x3.6	11.080	3.4672
8	TUBO-D108.0x3.6	11.810	3.6934
9	TUBO-D114.3x3.6	12.520	3.9161
10	TUBO-D127.0x4.0	15.450	4.3504
11	TUBO-D133.0x4.0	16.210	4.5629
12	TUBO-D139.7x4.0	17.050	4.8004
13	TUBO-D152.4x4.0	18.650	5.2483
14	TUBO-D159.0x4.0	19.480	5.4814
15	TUBO-D168.3x4.0	20.65	5.8102
16	TUBO-D193.7x4.5	26.75	6.6922
17	TUBO-D219.1x5.0	33.63	7.5716
18	TUBO-D244.5x5.4	40.56	8.4557
19	TUBO-D273.0x5.6	47.04	9.4570
20	TUBO-D298.5x5.9	54.23	10.3471
21	TUBO-D323.9x5.9	58.94	11.2450
22	TUBO-D355.6x6.3	69.13	12.3536
23	TUBO-D368.0x6.3	71.59	12.7895
24	TUBO-D406.4x6.3	79.19	14.1475
25	TUBO-D419.0x7.1	91.88	14.5645
26	TUBO-D457.2x7.1	100.4	15.9150

5. Formulation of Optimization Problem

It is shown that consideration of nonlinear behavior in the optimum design of structures not only provides more realistic results, but also produces lighter structures [5, 6]. Nonlinear structural behavior arises from a number of causes, which can be grouped into geometrical and material nonlinearity. If a structure experiences large deformations, its changing geometric configuration can cause the structure to respond nonlinearly. Nonlinear stress-strain relationships are a common cause of material nonlinear behavior. One of the main factors that can influence a material's stress-strain properties is load history in elastoplastic response.

In optimal design problem of linear and nonlinear scallop domes, the aim is to minimize the weight of the structure under stress and displacement constraints. This optimization problem can be expressed as follows.

$$\begin{aligned} &\text{Minimize } w(X) = \sum_{i=1}^{ng} x_i \sum_{j=1}^{nm} \gamma_j l_j, \\ &\text{subject to } g_{dj}(X) = \delta_j - \delta_{ju} \le 0, \quad j = 1, \ldots, p \\ &\qquad g_{sk}(X) = \sigma_k - \sigma_{ku} \le 0, \quad k = 1, \ldots, ne, \end{aligned} \tag{8}$$

where x_i, γ_j, and l_i are cross-sectional area of members belonging to group i, weight density, and length of jth element in this group, respectively; ng and nm are the total number of groups in the structure and the number of members in group n, respectively; δ_j and δ_{ju} are the displacement of jth joint and its upper bound, respectively; and p is the number of the nodes. Also σ_k and σ_{ku} are the stress of kth member and its upper bound, respectively; ne is the total number of members.

In the case of nonlinear optimization, design constraints include structural and stability constraints. The constraint ensuring the stability of the scallop dome during the optimization process is as follows:

$$g_s(X) = f_a - f_u \le 0, \tag{9}$$

where f_a is applied load and f_u is ultimate load determined using nonlinear analysis.

The allowable tensile and compressive stresses, in the case of linear analysis, are used according to the AISC ASD (1989) code [18] as follows:

$$\begin{aligned} \sigma_{ia} &= 0.6F_y \quad \text{for tensile stress,} \\ \sigma_{ia} &= F_{cr} \quad \text{for compression stress,} \end{aligned} \tag{10}$$

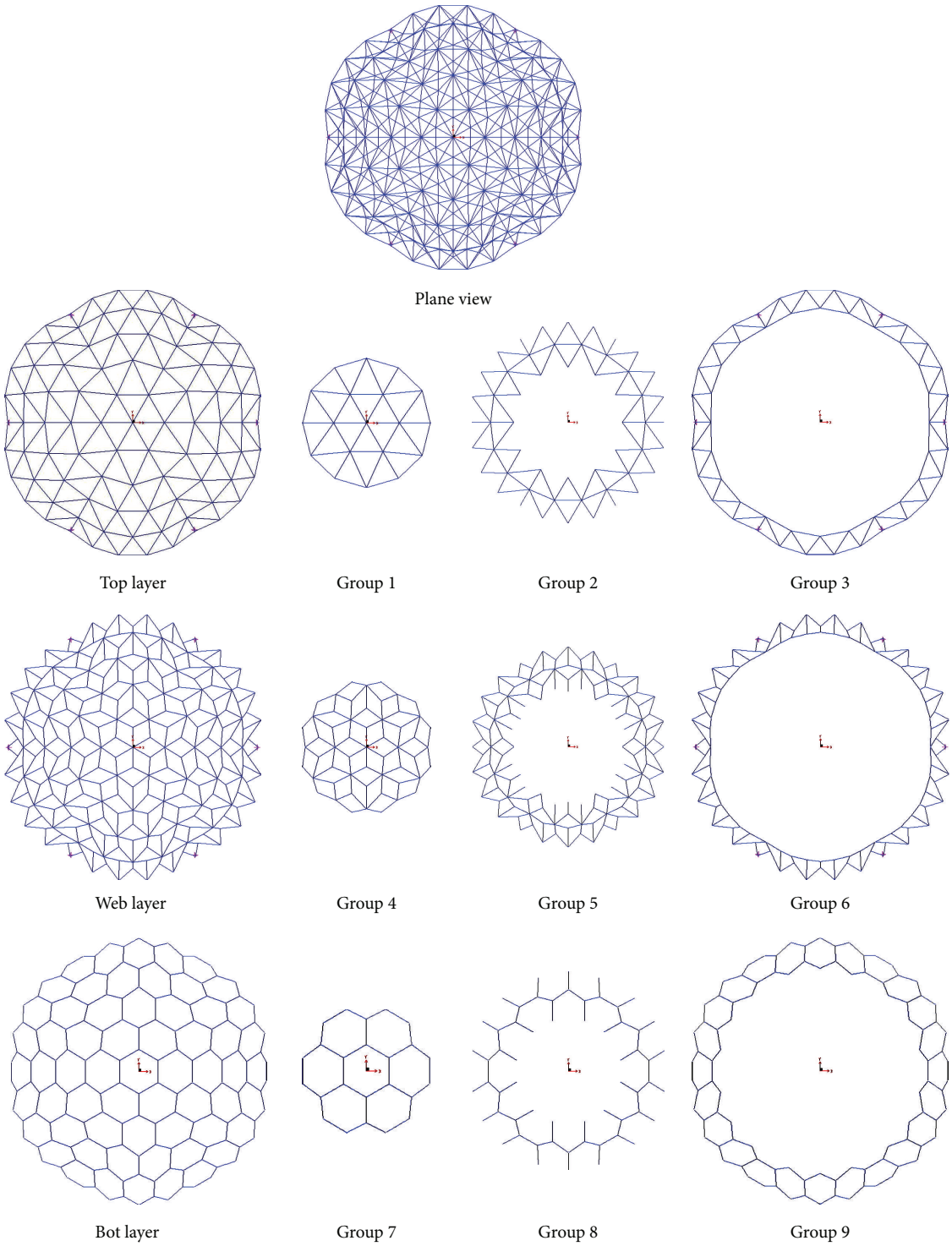


Figure 6: The 6-segment scallop dome with its relative element groups.

$$F_{cr} = \begin{cases} \dfrac{\left[\left(1 - \left(\lambda_i^2/2C_c^2\right)\right) F_y\right]}{\left[(5/3) + \left(3\lambda_i/8C_c\right) - \left(\lambda_i^3/8C_c^3\right)\right]} & \lambda_i < C_c, \\ \dfrac{12\pi^2 E}{23\lambda_i^2} & \lambda_i \geq C_c, \end{cases} \tag{11}$$

where E is the modulus of elasticity, F_y is the yield stress of steel, C_c is the slenderness ratio (λ_i) dividing the elastic and inelastic buckling regions ($C_c = \sqrt{2\pi^2 E/F_y}$), λ_i is the slenderness ratio ($\lambda_i = kl_i/r_i$), k is the effective length factor, l_i is the member length and r_i is the radius of gyration.

Table 2: Comparison of linear and nonlinear optimal designs for 6-segment double-layer scallop dome.

Design variables	Optimum design (cm^2)	
	Linear	Nonlinear
A_{Group1}	TUBO-D193.7x4.5	TUBO-D139.7x4.0
A_{Group2}	TUBO-D219.1x5.0	TUBO-D168.3x4.0
A_{Group3}	TUBO-D323.9x5.9	TUBO-D219.1x5.0
A_{Group4}	TUBO-D48.30x2.6	TUBO-D48.30x2.6
A_{Group5}	TUBO-D101.6x3.6	TUBO-D76.10x3.2
A_{Group6}	TUBO-D133.0x4.0	TUBO-D101.6x3.6
A_{Group7}	TUBO-D60.30x3.2	TUBO-D48.30x2.6
A_{Group8}	TUBO-D101.6x3.6	TUBO-D76.10x3.2
A_{Group9}	TUBO-D193.7x4.5	TUBO-D127.0x4.0
Weight (kg)	71994.14	44105.72
R	0.400	0.366
Overall time (min)	65.36	207.14
Maximum deflection (cm)	3.66	5.88

Table 3: Comparison of element stresses in optimal designs for 6-segment double-layer scallop dome.

Group no.	Linear stresses (kg/cm^2)				Nonlinear stresses (kg/cm^2)			
	σ^-	σ^-_{all}	σ^+	σ^+_{all}	σ^-	σ^-_{all}	σ^+	σ^+_{all}
1	−883.05	−1030.87	330.47	1440.00	−1353.55	−1483.97	538.04	2400.00
2	−1089.36	−1096.40	227.05	1440.00	−1745.17	−1774.71	367.48	2400.00
3	−1191.35	−1240.56	141.17	1440.00	−2022.98	−2031.79	218.55	2400.00
4	−274.93	−310.27	435.02	1440.00	−364.03	−594.68	548.77	2400.00
5	−668.37	−909.16	726.47	1440.00	−1027.25	−1177.98	1138.78	2400.00
6	−1091.87	−1122.73	623.65	1440.00	−1716.76	−1830.93	989.49	2400.00
7	−268.88	−485.61	132.90	1440.00	−457.95	−598.64	341.47	2400.00
8	−655.01	−893.69	122.29	1440.00	−1030.06	−1132.46	173.71	2400.00
9	−777.13	−1266.80	144.77	1440.00	−1411.50	−2080.07	309.72	2400.00

In the case of nonlinear analysis, the critical stress for a tension member is simply taken as the yield stress of steel. For a compression member, the safety factors can be dropped from both the above mentioned expressions [5]:

$$F_{cr} = \begin{cases} \left[\left(1 - \dfrac{\lambda_i^2}{2C_c^2}\right) F_y\right] & \lambda_i < C_c, \\ \dfrac{\pi^2 E}{\lambda_i^2} & \lambda_i \geq C_c. \end{cases} \tag{12}$$

In this study, penalty function method (PFM) is employed to handle the constraints of the structural optimization problem. PFM transforms the basic constrained optimization problem into alternative unconstrained one. The constrained optimization problem expressed by (8) can be converted into an unconstrained problem by constructing a function of the following form:

$$\Phi\left(X, r_p\right) = f(X) + p(X),$$
$$p(X) = r_p \sum_{i=1}^{ng} \left[\max\{0, g_i(X)\}\right]^2, \tag{13}$$

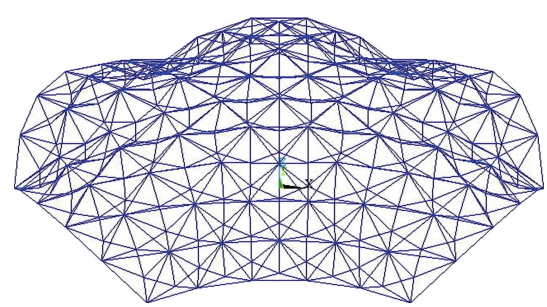


Figure 7: Deformed 6-segment nonlinear optimum double-layer scallop dome.

where Φ, p, and r_p are the pseudoobjective function, penalty function, and positive penalty parameter, respectively.

6. Firefly Algorithm

In structural optimization problems, where the objective function and the constraints are highly nonlinear functions of the design variables, the computational effort spent in gradient calculations required by the mathematical programming

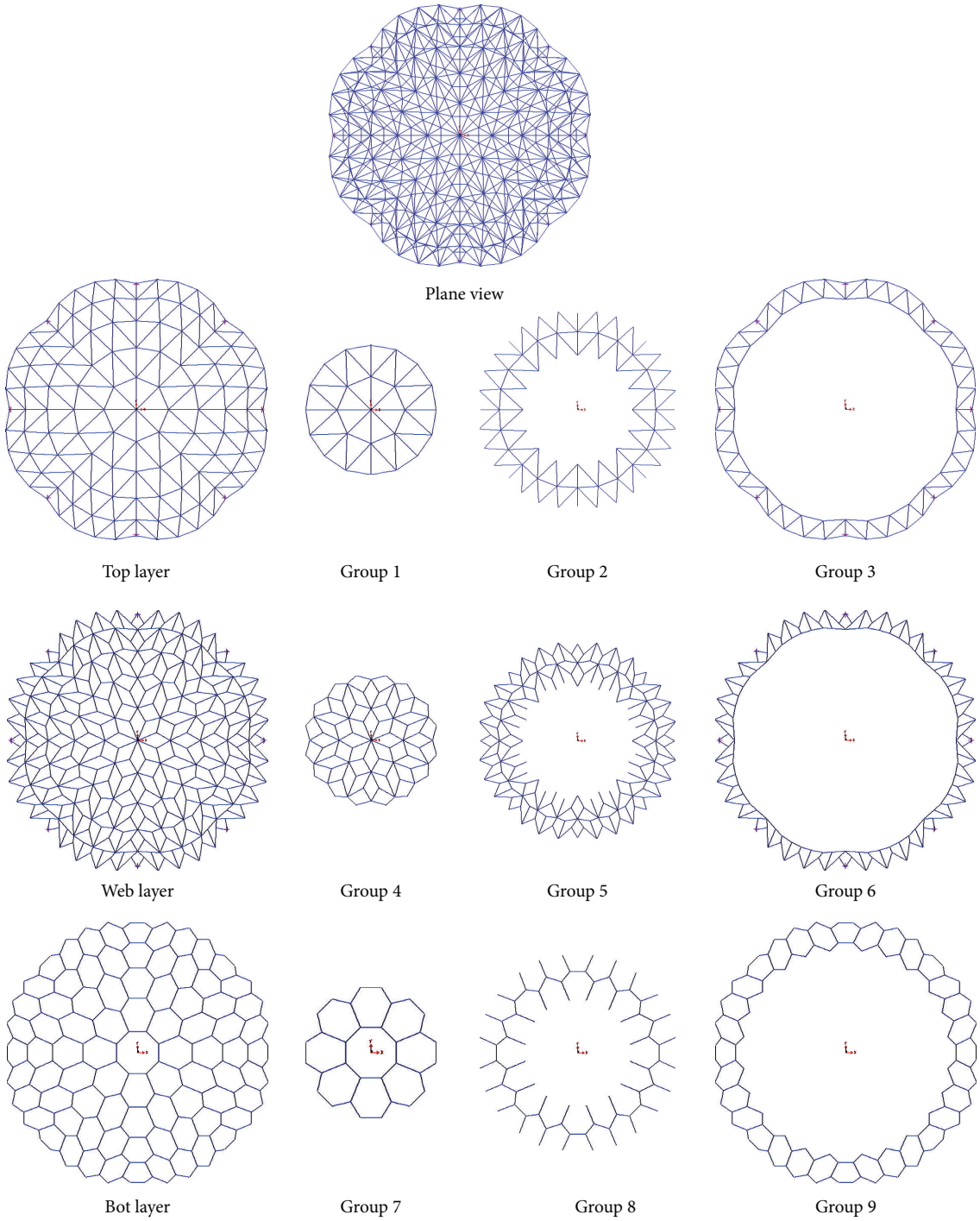


Figure 8: The 8-segment scallop dome with its relative element groups.

algorithms is usually large. In recent years, it was found that probabilistic search algorithms are computationally efficient even if greater number of optimization cycles is needed to reach the optimum. Furthermore, probabilistic methodologies were found to be more robust in finding the global optima, due to their random search, whereas mathematical programming algorithms may be trapped into local optima. During the recent years, many advanced met-heuristic optimization algorithms have been proposed by researchers [19–21]. However, in the present study, FA as a relatively new metaheuristic is used.

Gandomi et al. [22] employed FA for tackling benchmark complex optimization problems. Gholizadeh and Barati [23] compared the computational performance of particle swarm

Table 4: Comparison of linear and nonlinear optimal designs for 8-segment double-layer scallop dome.

Design variables	Optimum design (cm^2)	
	Linear	Nonlinear
A_{Group1}	TUBO-D159.0x4.0	TUBO-D133.0x4.0
A_{Group2}	TUBO-D168.3x4.0	TUBO-D133.0x4.0
A_{Group3}	TUBO-D219.1x5.0	TUBO-D168.3x4.0
A_{Group4}	TUBO-D60.30x3.2	TUBO-D60.30x3.2
A_{Group5}	TUBO-D88.90x3.2	TUBO-D60.30x3.2
A_{Group6}	TUBO-D168.3x4.0	TUBO-D114.3x3.6
A_{Group7}	TUBO-D76.10x3.2	TUBO-D60.30x3.2
A_{Group8}	TUBO-D82.50x3.2	TUBO-D101.6x3.6
A_{Group9}	TUBO-D152.4x4.0	TUBO-D114.3x3.6
Weight (kg)	60602.13	42557.28
R	0.364	0.326
Overall time (min)	60.05	198.98
Maximum deflection (cm)	3.04	3.97

Table 5: Comparison of element stresses in optimal designs for 8-segment double-layer scallop dome.

Group no.	Linear stresses (kg/cm^2)				Nonlinear stresses (kg/cm^2)			
	σ^-	σ^-_{all}	σ^+	σ^+_{all}	σ^-	σ^-_{all}	σ^+	σ^+_{all}
1	−843.37	−894.52	403.45	1440.00	−1009.92	−1386.13	489.38	2400.00
2	−937.10	−938.83	104.20	1440.00	−1298.67	−1386.13	212.35	2400.00
3	−1071.14	−1096.40	99.74	1440.00	−1759.16	−1774.71	146.34	2400.00
4	−455.26	−601.24	404.45	1440.00	−389.68	−1152.38	437.70	2400.00
5	−546.85	−889.45	789.40	1440.00	−655.55	−898.26	1133.44	2400.00
6	−945.21	−1234.81	518.78	1440.00	−1580.30	−2016.69	856.67	2400.00
7	−672.56	−863.53	117.78	1440.00	−817.19	−1165.20	164.46	2400.00
8	−696.54	−720.95	246.91	1440.00	−612.11	−1732.07	74.48	2400.00
9	−834.87	−1230.00	504.21	1440.00	−1289.62	−2076.34	751.59	2400.00

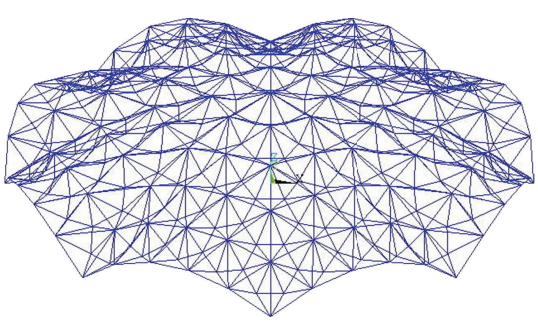


Figure 9: Deformed 8-segment nonlinear optimum double-layer scallop dome.

optimization (PSO), harmony search (HS), and FA algorithms for sizing and shaping optimisation of truss structures. Miguel et al. [24] employed FA for multimodal size, shape, and topology optimisation of truss structures.

The FA is a new metaheuristic optimization algorithm inspired by the flashing behavior of fireflies. FA is a population-based metaheuristic optimization algorithm. Fireflies communicate, search for prey, and find mates using bioluminescence with varied flashing patterns [22]. In order to develop the firefly algorithm, natural flashing characteristics of fireflies have been idealized using the following three rules [8].

(a) All of the fireflies are unisex; therefore, one firefly will be attracted to other fireflies regardless of their sex.

(b) Attractiveness of each firefly is proportional to its brightness; thus, for any two flashing fireflies, the less bright firefly will move towards the brighter one. The attractiveness is proportional to the brightness and they both decrease as their distance increases. If there is no brighter one than a particular firefly, it will move randomly.

(c) The brightness of a firefly is determined according to the nature of the objective function.

The attractiveness of a firefly is determined by its brightness or light intensity which is obtained from the objective function of the optimization problem. However, the attractiveness β, which is related to the judgment of the beholder, varies with the distance between two fireflies. The attractiveness β can be defined by [22]

$$\beta = \beta_0 e^{-\gamma \cdot r^2}, \tag{14}$$

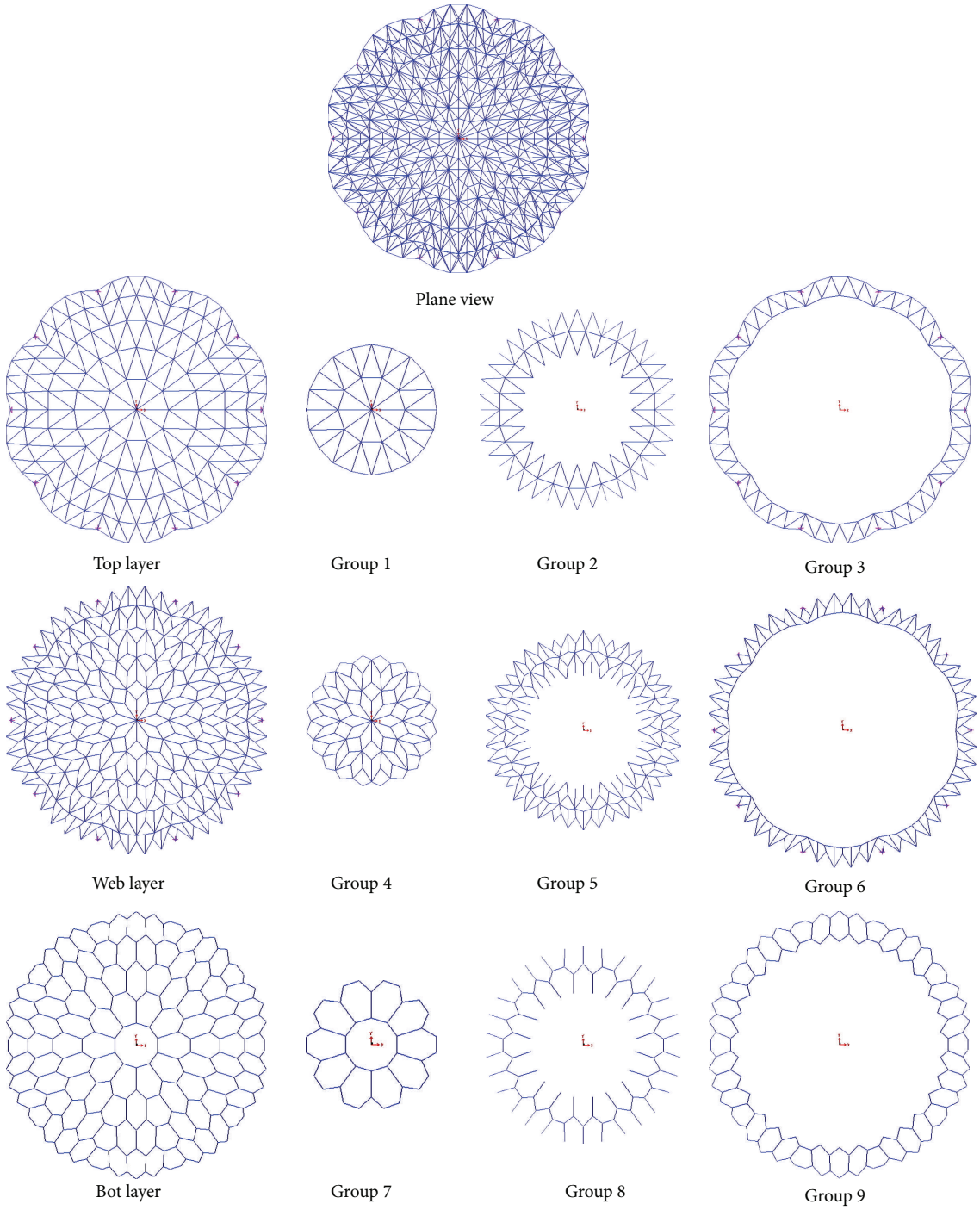


Figure 10: The 10-segment scallop dome with its relative element groups.

where r is the distance of two fireflies, β_0 is the attractiveness at $r = 0$, and γ is the light absorption coefficient.

The distance between two fireflies i and j at X_i and X_j, respectively, is determined using the following equation:

$$r_{ij} = \left\| X_i - X_j \right\| = \sqrt{\sum_{k=1}^{d} \left(x_{i,k} - x_{j,k} \right)^2}, \tag{15}$$

where $x_{i,k}$ is the kth parameter of the spatial coordinate x_i of the ith firefly.

In the FA, the movement of a firefly i towards a more attractive (brighter) firefly j is determined by the following equation:

$$X_i = X_i + \beta_0 e^{-\gamma \cdot r_{ij}^2} \left(X_j - X_i \right) + \alpha \left(\text{rand} - 0.5 \right), \tag{16}$$

Table 6: Comparison of linear and nonlinear optimal designs for 10-segment double-layer scallop dome.

Design variables	Optimum design (cm^2)	
	Linear	Nonlinear
A_{Group1}	TUBO-D152.4x4.0	TUBO-D127.0x4.0
A_{Group2}	TUBO-D159.0x4.0	TUBO-D127.0x4.0
A_{Group3}	TUBO-D193.7x4.5	TUBO-D152.4x4.0
A_{Group4}	TUBO-D60.30x3.2	TUBO-D48.30x2.6
A_{Group5}	TUBO-D76.10x3.2	TUBO-D60.30x3.2
A_{Group6}	TUBO-D133.0x4.0	TUBO-D88.90x3.2
A_{Group7}	TUBO-D76.10x3.2	TUBO-D60.30x3.2
A_{Group8}	TUBO-D76.10x3.2	TUBO-D60.30x3.2
A_{Group9}	TUBO-D88.90x3.2	TUBO-D82.50x3.2
Weight (kg)	57185.91	41272.77
R	0.380	0.340
Overall time (min)	61.22	201.1
Maximum deflection (cm)	2.53	3.29

where the second term is related to the attraction, while the third term is randomization with α being the randomization parameter. Also, rand is a random number generator uniformly distributed in $[0, 1]$.

In this paper, the modified equation proposed in [23] for computing α is employed as follows:

$$\alpha = \alpha_{\max} - \frac{\alpha_{\max} - \alpha_{\min}}{t_{\max}} \cdot t, \tag{17}$$

where $\alpha_{\max} = 1$ and $\alpha_{\min} = 0.2$. Also, $t_{\max}$ and t are the numbers of maximum iterations and present iteration, respectively.

7. Numerical Results

In the present work, three double-layer scallop domes with 6, 8, and 10 segments are considered. For all of the scallop domes, the span is 50.0 m, the height is 10 m, and the layer thickness is 1.5 m. The configurations of the mentioned scallop domes are shown in Figure 5. Young's modulus, mass density, and yield stress are 2.1×10^{10} kg/m^2, 7850 kg/m^3, and 2.4×10^6 kg/m^2, respectively. The computational time is measured in terms of CPU time of a PC Pentium IV 3000 MHz. A uniformly distributed load of 250 kg/m^2 is applied on the horizontal projection of the top layer.

In the present study, the number of fireflies is 20 and the maximum number of iterations is 500.

The ratio between the number of structural analyses actually required in the optimization process and the maximum number of structural analyses that would have been performed if all iterations were completed in each optimization cycle can be computed as follows:

$$R = \frac{np \times ng}{np \times mng}, \tag{18}$$

where nf, ng, and mng are the number of fireflies, number of generations, and maximum number of generations, respectively.

The allowable vertical deflection is 10 cm and the stress constraints are given in (10) to (11). The discrete design variables are selected from a set of standard pipe profiles listed in Table 1. In this table, cross-sectional area and radius of gyration are given by A and r, respectively.

For all examples, the structural elements of each layer are divided into three groups and therefore the optimization problem includes nine design variables:

$$X^T = \left\{A_{\text{Group1}}, A_{\text{Group2}}, A_{\text{Group3}}, A_{\text{Group4}}, A_{\text{Group5}}, A_{\text{Group6}}, A_{\text{Group7}}, A_{\text{Group8}}, A_{\text{Group9}}\right\}. \tag{19}$$

Example 1 (*A 6-segment double layer scallop dome*). The plane views of the 6-segment scallop dome together with its element groups are shown in Figure 6.

The optimization process considering linear and nonlinear behaviors is achieved and the results are given in Table 2.

In Table 3 the element stresses of the optimum design are compared with their corresponding allowable values and this comparison demonstrates that the solutions are feasible.

The nodal deflection of 6-segment optimum double-layer scallop dome found during the optimization process considering nonlinear behavior is represented in Figure 7.

Example 2 (*An 8-segment double-layer scallop dome*). The plane views of the 8-segment scallop dome together with its element groups are shown in Figure 8.

The optimization process considering linear and nonlinear behaviors is achieved and the results are given in Table 4.

In Table 5 the element stresses of the optimum designs are compared with their corresponding allowable values and this comparison demonstrates that the solutions are feasible.

The nodal deflection of 8-segment optimum double-layer scallop dome found during the optimization process considering nonlinear behavior is represented in Figure 9.

Table 7: Comparison of element stresses in optimal designs for 10-segment double-layer scallop dome.

Group no.	Linear stresses (kg/cm^2)				Nonlinear stresses (kg/cm^2)			
	σ^-	σ^-_{all}	σ^+	σ^+_{all}	σ^-	σ^-_{all}	σ^+	σ^+_{all}
1	−816.86	−858.66	366.93	1440.00	−999.69	−1284.66	423.84	2400.00
2	−847.26	−894.52	86.54	1440.00	−1067.89	−1284.66	115.28	2400.00
3	−1021.14	−1030.87	70.48	1440.00	−1630.34	−1633.65	123.01	2400.00
4	−503.74	−674.94	373.51	1440.00	−715.92	−836.82	525.53	2400.00
5	−483.55	−596.58	673.63	1440.00	−610.57	−700.99	862.08	2400.00
6	−923.48	−1169.62	482.38	1440.00	−1637.14	−1809.08	857.13	2400.00
7	−999.80	−1016.93	137.40	1440.00	−1260.07	−1594.17	157.69	2400.00
8	−468.43	−638.53	283.75	1440.00	−597.39	−750.56	373.16	2400.00
9	−1016.62	−1022.49	937.78	1440.00	−1146.79	−1833.77	877.02	2400.00

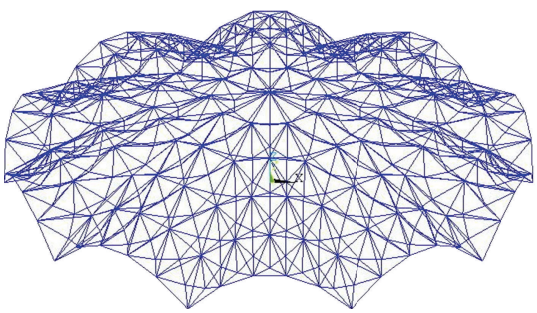


Figure 11: Deformed 10-segment nonlinear optimum double-layer scallop dome.

Example 3 (*A 10-segment double-layer scallop dome*). The plane views of the 10-segment scallop dome together with its element groups are shown in Figure 10.

The optimization process considering linear and nonlinear behaviors is achieved and the results are given in Table 6. In Table 7 the element stresses of the optimum designs are compared with their corresponding allowable values and this comparison demonstrates that the solutions are feasible.

The nodal deflection of 10-segment optimum double-layer scallop dome found during the optimization process considering nonlinear behavior is represented in Figure 11.

8. Conclusions

The present study deals with size design optimization of scallop domes for static loading. The cross-sectional areas of the element groups are the design variables and the weight of the structure is the objective function of the optimization problem. The design variables are selected from a set of available standard sections; consequently, the optimization problem is discrete. Two optimization processes considering linear and nonlinear behavior of the structure are included. In the nonlinear optimization processes, geometrical nonlinearity is involved. In both optimization process, stress and deflection constraints are checked, but in the nonlinear optimization process the safety factors are dropped from the stress check equations. In order to implement the optimization, FA is employed. Considering 6-, 8- , and 10-segment scallop domes with nine element groups the optimization processes are achieved. In all examples the weight of nonlinear structure is significantly less than that of linear one. Also it is observed that in these examples, the number of required generations by the nonlinear optimization process is less than that of the linear one. The numerical results demonstrate that by taking into account the nonlinear behavior a significant reduction in the optimum weight of the scallop domes can be achieved compared with those of obtained involving linear behavior. Also, it is observed that by increasing the number of segments the weight of linear and nonlinear structures is decreased.

References

[1] H. Nooshin, H. Tomatsuri, M. Fujimoto, and S. Domes, *IASS97 Symposium on Shell & Spatial Structures: Design, Performance & Economics*, CI-Premier Pte Ltd, Singapore, 1997.

[2] E. Salajegheh and S. Gholizadeh, "Optimum design of structures by an improved genetic algorithm using neural networks," *Advances in Engineering Software*, vol. 36, no. 11-12, pp. 757–767, 2005.

[3] A. Kaveh, B. Farhmand Azar, and S. Talatahari, "Ant colony optimization for design of space trusses," *International Journal of Space Structures*, vol. 23, no. 3, pp. 167–181, 2008.

[4] V. Toğan and A. T. Daloğlu, "Optimization of 3d trusses with adaptive approach in genetic algorithms," *Engineering Structures*, vol. 28, no. 7, pp. 1019–1027, 2006.

[5] M. P. Saka and M. Ulker, "Optimum design of geometrically nonlinear space trusses," *Computers and Structures*, vol. 41, no. 6, pp. 1387–1396, 1991.

[6] R. Kamyab Moghdas and E. Salajegheh, "Size optimization of nonlinear scallop domes by an enhanced particle swarm algorithm," *International Journal of Civil Engineering*, vol. 11, pp. 77–89, 2013.

[7] *ANSYS Release 10.0 Documentation*, ANSYS, Inc., Houston, Tex, USA, 2006.

[8] X. S. Yang, "Firefly algorithms for multimodal optimization," in *Stochastic Algorithms: Foundations and Applications*, O. Watanabe and T. Zeugmann, Eds., vol. 5792 of *Lecture Notes in Computer Science*, pp. 169–178, Springer, Berlin, Germany, 2009.

[9] S. Talatahari, "Determining the optimum section of tunnels using ant colony optimization," *Mathematical Problems in Engineering*, vol. 2013, Article ID 320360, 7 pages, 2013.

[10] S. Gholizadeh, A. Pirmoz, and R. Attarnejad, "Assessment of load carrying capacity of castellated steel beams by neural networks," *Journal of Constructional Steel Research*, vol. 67, no. 5, pp. 770–779, 2011.

[11] A. H. Gandomi, S. Talatahari, X. S. Yang, and S. Deb, "Design optimization of truss structures using cuckoo search algorithm," *The Structural Design of Tall and Special Buildings*, vol. 22, no. 17, pp. 1330–1349, 2013.

[12] S. Talatahari, M. Kheirollahi, C. Farahmandpour, and A. H. Gandomi, "A multi-stage particle swarm for optimum design of truss structures," *Neural Computing and Applications*, vol. 23, no. 5, pp. 1297–1309, 2013.

[13] S. Gholizadeh and O. A. Samavati, "Structural optimization by wavelet transforms and neural networks," *Applied Mathematical Modelling*, vol. 35, no. 2, pp. 915–929, 2011.

[14] R. Kamyab and E. Salajegheh, "Prediction of nonlinear time history deflection of scallop domes by neural networks," *International Journal of Optimization in Civil Engineering*, vol. 1, pp. 419–432, 2011.

[15] R. Kamyab Moghadas, C. K. Keong, and S. B. Mohd, "Prediction of optimal design and deflection of space structures using neural networks," *Mathematical Problems in Engineering*, vol. 2012, Article ID 712974, 18 pages, 2012.

[16] *MATLAB—The Language of Technical Computing*, The Math Works, Inc., Natick, Mass, USA, 2006.

[17] M. A. Crisfield, *Non-Linear Finite Element Analysis of Solids and Structures*, vol. 1, John Wiley & Sons, Chichester, UK, 1991.

[18] AISC, *Manual of Steel Construction-Allowable Stress Design*, American Institute of Steel Construction, Chicago, Ill, USA, 9th edition, 1989.

[19] S. Gholizadeh, "Layout optimization of truss structures by hybridizing cellular automata and particle swarm optimization," *Computers and Structures*, vol. 125, pp. 86–99, 2013.

[20] S. Gholizadeh and A. Barzegar, "Shape optimization of structures for frequency constraints by sequential harmony search algorithm," *Engineering Optimization*, vol. 45, no. 6, pp. 627–646, 2013.

[21] A. Kaveh and S. Talatahari, "Particle swarm optimizer, ant colony strategy and harmony search scheme hybridized for optimization of truss structures," *Computers and Structures*, vol. 87, no. 5-6, pp. 267–283, 2009.

[22] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Mixed variable structural optimization using Firefly Algorithm," *Computers and Structures*, vol. 89, no. 23-24, pp. 2325–2336, 2011.

[23] S. Gholizadeh and H. Barati, "A comparative study of three metaheuristics for optimum design of trusses," *International Journal of Optimization in Civil Engineering*, vol. 2, pp. 423–441, 2012.

[24] L. F. F. Miguel, R. H. Lopez, and L. F. F. Miguel, "Multimodal size, shape, and topology optimisation of truss structures using the Firefly algorithm," *Advances in Engineering Software*, vol. 56, pp. 23–37, 2013.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 680359, 13 pages
http://dx.doi.org/10.1155/2013/680359

Research Article

Engineering Knowledge-Based Variance-Reduction Simulation and G-Dominance for Structural Frame Robust Optimization

D. Greiner, J. M. Emperador, B. Galván, M. Méndez, and G. Winter

Instituto de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería (SIANI), Universidad de Las Palmas de Gran Canaria, 35017 Las Palmas de Gran Canaria, Spain

Correspondence should be addressed to D. Greiner; dgreiner@iusiani.ulpgc.es

Received 8 August 2013; Revised 15 October 2013; Accepted 16 October 2013

Academic Editor: Amir H. Gandomi

This paper proposes the incorporation of engineering knowledge through both (a) advanced state-of-the-art preference handling decision-making tools integrated in multiobjective evolutionary algorithms and (b) engineering knowledge-based variance-reduction simulation as enhancing tools for the robust optimum design of structural frames taking uncertainties into consideration in the design variables. The simultaneous minimization of the constrained weight (adding structural weight and average distribution of constraint violations) on the one hand and the standard deviation of the distribution of constraint violation on the other are handled with multiobjective optimization-based evolutionary computation in two different multiobjective algorithms. The optimum design values of the deterministic structural problem in question are proposed as a reference point (the aspiration level) in reference-point-based evolutionary multiobjective algorithms (here g-dominance is used). Results including *S*-metric statistics in a structural frame test case with uncertain loads show considerable reductions in computational costs without harming the nondominated front quality, obtaining a design set that makes it possible to select minimum weight and maximum robustness optimum designs.

1. Introduction

Evolutionary algorithms have been applied since their origins [1] in the mechanical and structural optimization field. Their global optimization nature, derived from the fact that they are based on population searches, enables them to avoid the stagnation in frequent local minima that occurs in structural problems. In addition, they permit discrete optimization, matching real requirements in metallic bar structures. They have been applied in bar structures, trusses, and frames, as well as in metallic or concrete structures; surveys of the state of the art can be found in [2–4]. Their principal aim has been focused on the weight minimization problem, which is directly related with the raw material cost. It is an ongoing active field in mechanical and structural design (see e.g., [5]).

Evolutionary multiobjective algorithms have been developed since the mid nineteen eighties, but their success in a wide range of science and engineering fields has really taken off since the late nineties [6]. These population-based global search algorithms also have extensive applications in the structural design field, for example, in [7–9]. The incorporation of engineer/decision maker preferences in the evolutionary multiobjective optimization process has been a topic of increasing interest in recent years (e.g., [10–13]). This is particularly so, when engineer/decision maker preference information is considered to constitute desirable aspiration levels for objective functions; the use of a reference point when using evolutionary multiobjective computation has led to various recent proposals: Ben Said et al. propose the r-dominance in [14], Deb et al. propose the r-NSGAII in [15], and Thiele et al. [16] and Molina et al. propose the g-dominance in [17].

There has been a general move to include uncertainties in optimum engineering design and this has played a key role in the field of multidisciplinary design optimization (MDO) [18]. In particular, in the field of structural design optimization, two approaches are considered: reliability-based design optimization (RBDO) and robust design optimization (RDO) (see [19–21]). Some recent research work in structural engineering design is by Coelho [22] and by Greiner and Hajela [23].

A review of structural optimization for robust design [24] can be found by Doltsinis et al. [25], where robust truss design is performed by optimization of the mean and variance of the fitness function considering a weighted approach for optimization and a Taylor series approach for simulation. A Monte-Carlo simulation coupled with a hybrid optimization is carried out by a single-objective genetic algorithm for topology identification and conventional nonlinear programming for continuous cross-section design; reducing the response of fluctuations is performed by Sandgren and Cameron [26]. Structural truss robust design has been handled by Lagaros et al. in [27, 28] considering multiobjective optimization evolutionary methods with a priori articulation of preferences (cascade evolutionary algorithm), for simultaneous minimization of the weight and response (characterized by one point displacement), as well as in [29, 30]. A review of robust optimal design in dynamics is also done by Zang et al. [31].

Although other approaches are possible, Monte-Carlo simulations are still useful tools when dealing with uncertainties in multiobjective design [22]. A structural robust design optimization of frames with engineering knowledge-based variance-reduction simulation that allowed a high reduction in computational cost in terms of the required number of fitness function evaluations versus a previous conventional approach was proposed by Greiner et al. [32]. This paper proposes to combine that approach with the consideration of the engineer/decision maker's preference information as desirable aspiration levels for objective functions (reference point(s)) when using evolutionary multiobjective computation. These aspiration levels are materialized in the case of robust design as the values of the fitness function of the optimum design of the deterministic problem (without considering uncertainties) as well as zero variation of the performance function, as proposed by Greiner et al. [33]. In this paper we introduce a methodology that consists of the combination of preference-based multiobjective evolutionary algorithms introducing aspiration levels as a reference point; here, the g-dominance is used, with engineering knowledge variance reduction in the Monte-Carlo simulation process for improving computational performance of robust design in structural engineering.

The structure of the paper is as follows. In Section 2 the structural problem addressed is presented, introducing first the deterministic problem and later a design dealing with uncertainties. Section 3 briefly describes evolutionary multiobjective algorithms and the incorporation of preference information as aspiration levels with reference point information, used in the optimization procedure. Later, Section 4 explains the structural frame test case considered. Then, Section 5 gives the results and discussion, including a comparison between the standard and proposed reduced simulation procedures. Finally, the paper ends with the conclusions section.

2. The Structural Problem

2.1. Deterministic Design. The fitness function, in order to perform the constrained weight minimization, has to take into account those proper requirements of the bar structure to perform its mission. Its value is directly related with the acquisition cost of raw material of the metallic frame. The information needed by the fitness function is obtained through the direct stiffness method and the following constraints are applied, in order to guarantee the appropriate functionality of the structure.

(a) *Stresses of the bars*, where the stress limit depends on the frame material (e.g., common construction steel has a stress limit of 2600 kgp/cm^2) and the stress compared takes into account the axial and shearing stresses by the shear effort and also the bending effort, using Von Misses yield criteria. For each bar, (1) has to be accomplished. Consider

$$\sigma_{\text{co}} - \sigma_{\text{lim}} \leq 0, \tag{1}$$

where σ_{co} refers to the comparing stress (maximum stress of the bar) and σ_{lim} refers to the limit stress.

(b) *Compressive slenderness limit*, for each bar where the buckling effect is considered (the value in question may vary, depending on the particular code used) (2) has to be satisfied. Consider

$$\lambda - \lambda_{\text{lim}} \leq 0, \tag{2}$$

where λ refers to the slenderness compared (compressive slenderness of the bar) and λ_{lim} refers to the limit slenderness.

(c) *Displacements of joints or middle points* of bars are also possible requirement, as observed in (3). Consider

$$u_{\text{co}} - u_{\text{lim}} \leq 0, \tag{3}$$

where u_{co} refers to the displacement compared and u_{lim} refers to the limit displacement.

With these constraints, the fitness function *constrained weight* integrates the constraints violations as weight penalties, as shown in

$$\text{Fitness Function} = \left[\sum_{i=1}^{N_{\text{bars}}} A_i \cdot \rho_i \cdot l_i\right]\left[1 + k \cdot \sum_{j=1}^{N_{\text{viols}}} \left(\text{viol}_j - 1\right)\right], \tag{4}$$

where A_i is the area of the bar section type i; ρ_i is the density of bar i; l_i is the length of bar i; k is the constant that regulates the equivalence between weight and restriction (suitable values around one); viol_j, for each violated restriction j, is the quotient between the violated restriction value (stress, displacement, or slenderness) and its reference limit.

2.2. Robust Design

2.2.1. Introduction. The deterministic optimum design of a bar structure is frequently defined by the constraints imposed

in terms of stress, displacement, or buckling, which are taken up to, but not over, their limit values.

In real structures, loads often vary and this is taken into consideration in the design codes. So, a deterministic optimized structure, in which the corresponding constraints come close to limit values, is expected to be more sensitive to those random variations. An analysis of those uncertainties is required to guarantee a robust design. The principal objective of robust design is to find a solution in which structural performance is less sensitive to the fluctuations of parameters without eliminating their variation.

The variation in load actions that act on a structure from the viewpoint of probabilistic or semiprobabilistic safety criteria is related to the consideration of loads as random variables and with the existence of some ultimate limit states that lead to total or partial ruin of the structure and some serviceability limit states that they, when reached, produce malfunction [34–36]. In order to define the actions, it is assumed here that the corresponding variation follows a Gaussian probability density function. The characteristic value of an action is defined as the value that belongs to the 95% percentile; that is, the probability of it being exceeded is 0.05 (see (5)). Consider

$$F_k = F_m (1 + k \cdot \delta), \tag{5}$$

where F_k is the characteristic value, F_m is the average value, k is the coefficient corresponding to the percentile of 5% = 1.64, and δ is the typical deviation of the average value expressed per unit variation. A structure is usually calculated taking into account the characteristic values of the actions, which are defined based on said actions and depending on the specific national code in question.

2.2.2. Problem Definition. The mathematical formulation of the robust design problem is as follows.

One individual is a structure E of cross sections $(\mathrm{CS}_1, \mathrm{CS}_2, \mathrm{CS}_3, \ldots, \mathrm{CS}_r)$, r being the total number of bars in the structure, with weight w and statistical parameters $(\mathrm{Constraint}_E, \sigma_{\mathrm{Const}_E})$ that are the average and standard deviation, respectively, of the constraint violations (the total number of constraint violations per structure is numviol), as shown in

$$\mathrm{Constraint}_E = \frac{\sum_1^m \sum_{j=1}^{N_{\mathrm{viol}}} \mathrm{viol}_j}{m}, \tag{6}$$

$$\sigma_{\mathrm{Constraint}_E} = \sqrt{\frac{\sum_1^m \sum_{j=1}^{N_{\mathrm{viol}}} \left(\mathrm{viol}_j - \mathrm{Constraint}_E\right)^2}{m-1}}, \tag{7}$$

where m is the number of Monte-Carlo simulation samples. In this paper, m is evaluated by two different procedures.

(a) Standard Monte-Carlo: $m = 30^n$, n being the number of random variables.

(b) Reduced procedure (engineering knowledge-based variance-reduction Monte-Carlo): $m = \mathrm{card}(Q)$, Q being the set of samples such that

$$\begin{gathered} F_1 > F_{k1} \text{ OR } F_2 > F_{k2} \text{ OR } F_3 > F_{k3} \\ \text{OR } \cdots F_i > F_{ki} \cdots \text{OR } F_n > F_{kn}. \end{gathered} \tag{8}$$

F_i is the effective applied load and F_{ki} the characteristic value; those samples belonging to m are the only cases proposed to be evaluated by the finite element code; the contribution of the rest of the cases to the constraint violation distribution is estimated to be zero; therefore, they do not need to be evaluated by the finite element code with the corresponding computational cost saving [32].

Multiobjective robust design problem (as opposed to the formulation of deterministic design that does not take uncertainties into consideration, shown in (4)) consists in finding E such that (9) and (10) hold. Consider

$$\begin{aligned} &\text{Min Fitness Function 1} \\ &= \left[\sum_{i=1}^{N_{\mathrm{bars}}} A_i \cdot \rho_i \cdot l_i\right] \left[1 + k \cdot \sum_{j=1}^{N_{\mathrm{viols}}} \left(\mathrm{Constraint}_E - 1\right)\right], \end{aligned} \tag{9}$$

$$\text{Min Fitness Function 2} = \sigma_{\mathrm{Constraint}_E}. \tag{10}$$

3. Evolutionary Multiobjective Algorithms and Preferences Inclusion through Reference Point

Here, the g-dominance approach (as a method of considering desirable aspiration levels in evolutionary computation) has been used in two evolutionary multiobjective algorithms: NSGAII [37] and SPEA2 [38], although the methodology would be equally valid using other approaches with similar characteristics. G-dominance enhances those solutions that either (a) dominate the reference point or (b) are dominated by the reference point, discriminating all the others. So, when we are interested in the environment of a certain point in the objective space, it is a very useful tool. From the perspective of decision making, the preference information of the decision maker is given here as the reference point coordinates. Two ways of implementing g-dominance are proposed in [17]: (a) changing the dominance-checking function and (b) changing the evaluation of the objective functions by greatly penalizing those solutions which do not belong to the area of interest as described previously. Here, the latter option (in [17] claimed to be the simplest) is used.

The use of g-dominance for robust optimization of structural bars was first introduced by Greiner et al. [33]. In our structural problem, there is engineering knowledge that can help in defining our aspiration levels (i.e., our reference point in the g-dominance). When no uncertainties are taken into account, the optimum design is a deterministic one,

and when constraints are present, it should satisfy all of them (i.e., no constraint violations are allowed). By contrast, when some of the external variables/parameters are random variables, we are dealing with uncertainties. The above-mentioned deterministic optimum design often lies on the boundary of the constraints violation and when considering the external random variables, then it is highly probable that this deterministic optimum violates the constraints in this uncertain environment. In our structural frame optimization problem, a robust design implies the achievement of designs with (1) average low constrained weight (good performance) and (2) low constraint violation standard deviations (sensitivity to external variations is minimum: zero variation in the deterministic case). Both 1 and 2 are two simultaneous minimization objectives. Therefore, we consider the function values of the deterministic optimum design as estimators of a lower limit of the solutions when considering uncertainties in the optimization, because it is expected that those deterministic optimum values dominate the designs of the problem when considering random variables. That is, those values are the aspiration levels considered as a reference point in the g-dominance criterion.

Therefore, the deterministic optimum design fitness function values are taken as the reference point in g-dominance for multiobjective optimization when considering uncertainties in the design. The advantages of combining this approach with the engineering knowledge-based variance reduction simulation in terms of computational time reduction are shown in the results section.

4. Test Case

The reference test case considered is based on a problem taken from [39] for single objective weight minimization using continuous variables. The solution reported in the previous reference using classical optimization methods was improved using evolutionary algorithms from [40, 41], where the discrete optimization problem using real cross-section types and including the buckling effect was also solved. This last deterministic evolutionary optimum design is taken as a reference in this work and compared with the robust optimum design nondominated front.

The test case is depicted in Figure 1, where the spot lengths (10 and 20) and height (6) are in meters, the loads in kN/m, 14.7, 9.8, and 1.96, and there is a constraint of maximum displacement of middle point of bar 2 equal to length/300, that is, $6.67 \cdot 10^{-2}$ m. It is a discrete domain problem, belonging the cross-section types to the European IPE class (16 different types per bar). The buckling effect and its own gravitational load have been taken into account. The density considered ($7.85 \cdot 10^{-3}$ kg/m^3) and Young modulus ($205.8 \cdot 10^9$ Pa) are common steel values and the yield stress is $235.2 \cdot 10^6$ Pa (the structure is assumed to be in elastic behaviour).

In addition, a second test case is analysed (incremented loading test case), where all the loads are increased by 5% in value, maintaining the rest of the characteristics as in the previous test case.

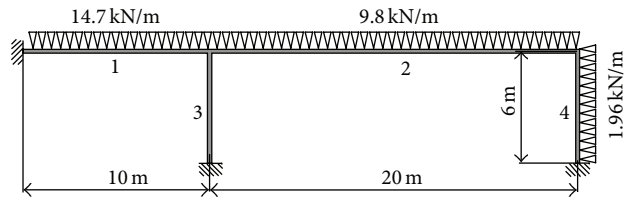


Figure 1: Frame test case with four bars.

5. Results and Discussion

Ten independent executions were performed for each multiobjective evolutionary algorithm. Specifically, multiobjective genetic algorithms (both NSGAII and SPEA2) were used with a population size of 200 individuals, uniform crossover, and uniform mutation rate of 0.06 [42]. A stop criterion of 100 generations was considered in all cases.

5.1. Standard Procedure. The standard Monte-Carlo simulation was performed considering 30^N simulations per structural design in order to construct the constraint violation distribution, N being the number of different variables considered. Here, the simulated variables correspond to the three linear uniform loads of the frame structure, one for each loaded bar.

The distribution of each linear uniform load is simulated through a Gaussian distribution, which is calculated considering the test case load value to be the characteristic value. Therefore, three independent Gaussian random variables are considered: the loads applied on bars 1 and 2 have been assigned a lower standard deviation than the load applied on bar 3, which could be assimilated as wind load and has been assigned a higher variation. Their distributions are represented graphically in Figure 2.

Optimization results are shown by Greiner et al. [43] and graphically combined in Figure 3 (NSGAII and SPEA2), which shows the final nondominated front evaluated from the accumulated total number of executions performed using both algorithms. The x-axis belongs to the constrained weight value (in kg), obtained by adding the weight of the particular structural design and the average of the constraints violation distribution in terms of weight, where the constraint violation is evaluated as in (6). The y-axis belongs to the typical deviation of the constraints violation distribution in terms of weight.

A total of twelve different frame structural designs make up the Pareto optimal front obtained. They are detailed graphically in Figure 3. Due to the stochastic character of Monte-Carlo simulations, it is possible to achieve different values of the objective functions for a single design. However, the differences between them are minor, indicating that the system simulation has been suitably performed.

The two algorithms (NSGAII and SPEA2) perform similarly and are capable of locating the extreme frame structural design solutions. The number of final Pareto front designs located is similar when the nondominated front obtained by accumulating the whole solution set is taken into account: 14 in NSGAII and 12 in SPEA2. Both algorithms also found

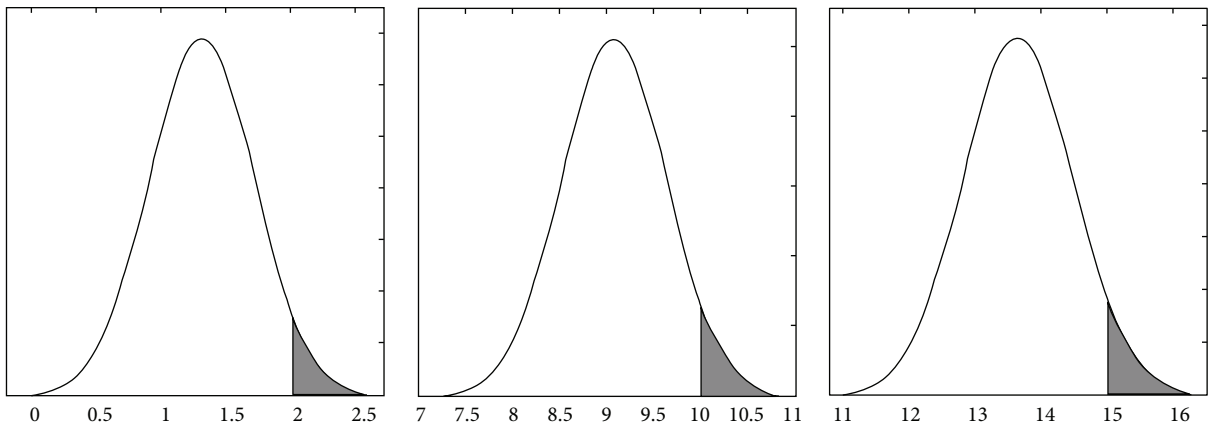


Figure 2: Test case load distributions, with shaded zones corresponding to the values greater than the characteristic load (in $10^{-1} \cdot$ T/m), equivalent to 1.96, 9.8, and 14.7 kN/m, respectively.

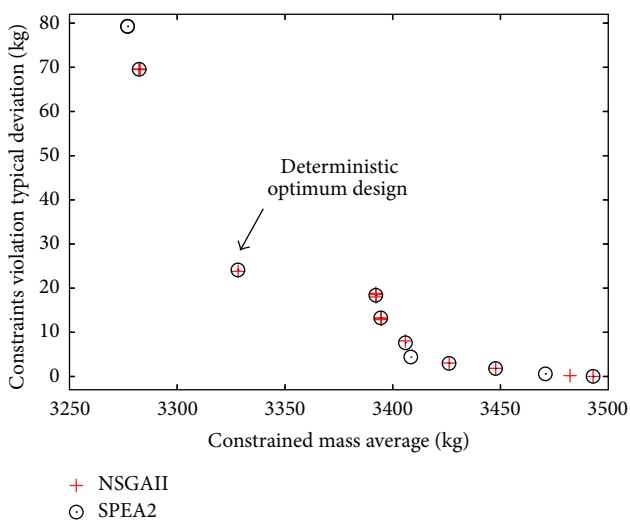


Figure 3: Final nondominated front obtained by accumulated SPEA2 and NSGAII executions (constrained weight versus constraint violation typical deviation, kg) [43].

all the twelve design solutions, although the constraint variations caused by the Monte-Carlo simulation produce the effect that some of these points are dominated and not shown in the final front results (Figure 3).

The deterministic optimum design, whose loads have the characteristic value of the imposed Gaussian distributions, has no constraint violations and a weight of 3324.3 kg. Its corresponding design for uncertain loads is highlighted in bold type in Table 1 and also arrowed in Figure 7. When the robust design including the load variations is considered this design is seen to violate the constraints on certain occasions, with a standard deviation and mean distribution of 24.1 kg and 3.8 kg, respectively. Therefore, the engineer or decision maker should select an individual among this deterministic optimum design and the solution located furthest to the right of the front, which has no constraint violations at all, even when uncertainties are taken into consideration (corresponding to a design weight of 3492.9 kg and 5.1% higher in terms of weight; see Table 1).

5.2. Reduced Procedure. The standard Monte-Carlo simulation evaluates $30^3 = 27000$ structures for each evolutionary computation solution candidate in our test case. Here, the variance reduction procedure inspired by the well-known importance sampling Monte-Carlo technique [44] and based on engineering design knowledge is used to reduce the computational cost without losing solution quality. It was first introduced by Greiner et al. [32], where the NSGAII algorithm was used as an optimizer.

Structural evaluations are required to calculate the possible constraints produced by each load case and hence their contribution to the constraint violation distribution. Its standard deviation is chosen as the measure of structural robustness of each design; the structure with the lowest weight and zero standard deviation corresponds to the solution located further to the right of the Pareto frontier; by contrast, the structure with the highest typical deviation and lowest weight corresponds to the solution located on the far left of the Pareto frontier.

Considering the constant load of the deterministic optimization problem as the characteristic value (i.e., the value belonging to the 95% percentile) of the Gaussian model in the problem that handles uncertainties, the reduced procedure (as explained in Section 2.2.2) proposes evaluation only of those load cases that surpass at least one of its characteristic values (1.96, 9.8, and 14.7 kN/m, as shown in Figure 2) for each structural design. Therefore, it is estimated that only $1 - 0.95^3 = 0.142625$, (14.2625% in percentage) of the load cases will be evaluated using this reduced simulation procedure, with the proportional benefit in terms of computational cost (here, only 3,850 structural design evaluations out of 27,000). The rest of the cases are assigned a null contribution to the constraint distribution. Here, results corresponding with the SPEA2 multiobjective optimizer are obtained and shown in Figure 4. They are very similar to the results obtained with the NSGAII in [32].

A total of thirty-three different frame structural designs make up the Pareto optimal front obtained, as had previously been achieved using NSGAII by Greiner et al. [32]. They are detailed graphically in Figure 5, where NSGAII and

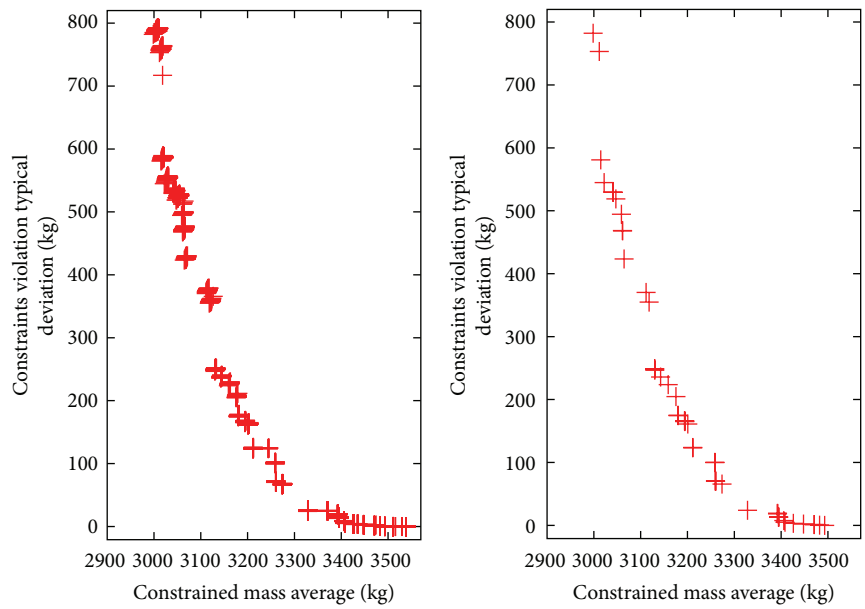


Figure 4: Accumulated nondominated fronts of each of the ten independent executions by SPEA2 (a) and their nondominated solutions (b). Reduced distribution.

Table 1: Detailed nondominated structural frame designs of Figure 7.

Constrained weight (kg)	Constraint violation typical deviation (kg)	Constraint violation average (kg)	Cross section type bar 1	Cross section type bar 2	Cross section type bar 3	Cross section type bar 4	Obtained with the EMO algorithm
3328.15	24.146	3.834	**IPE330**	**IPE500**	**IPE450**	**IPE500**	SPEA2
3328.26	24.002	3.946	,,	,,	,,	,,	NSGAII
3328.27	23.806	3.955	,,	,,	,,	,,	NSGAII
3392.12	18.823	2.335	IPE400	IPE550	IPE220	IPE450	NSGAII
3392.13	18.182	2.347	,,	,,	,,	,,	NSGAII
3392.21	18.018	2.427	,,	,,	,,	,,	NSGAII
3394.47	13.24	1.234	IPE360	IPE550	IPE300	IPE450	SPEA2
3394.52	12.9	1.277	,,	,,	,,	,,	SPEA2
3394.53	12.822	1.293	,,	,,	,,	,,	SPEA2
3405.85	7.673	0.517	IPE330	IPE500	IPE500	IPE500	SPEA2
3405.86	7.63	0.533	,,	,,	,,	,,	SPEA2
3408.36	4.407	0.207	IPE400	IPE550	IPE160	IPE500	SPEA2
3426.16	3.203	0.11	IPE400	IPE550	IPE180	IPE500	SPEA2
3426.17	3.047	0.114	,,	,,	,,	,,	NSGAII
3426.18	2.972	0.129	,,	,,	,,	,,	SPEA2
3447.76	1.861	0.041	IPE400	IPE550	IPE200	IPE500	NSGAII
3447.77	1.816	0.046	,,	,,	,,	,,	SPEA2
3470.81	0.638	0.011	IPE400	IPE550	IPE220	IPE500	SPEA2
3482.26	0.202	0.003	IPE360	IPE550	IPE360	IPE450	NSGAII
3492.94	0	0	IPE400	IPE550	IPE160	IPE550	NSGAII
3492.94	0	0	,,	,,	,,	,,	SPEA2

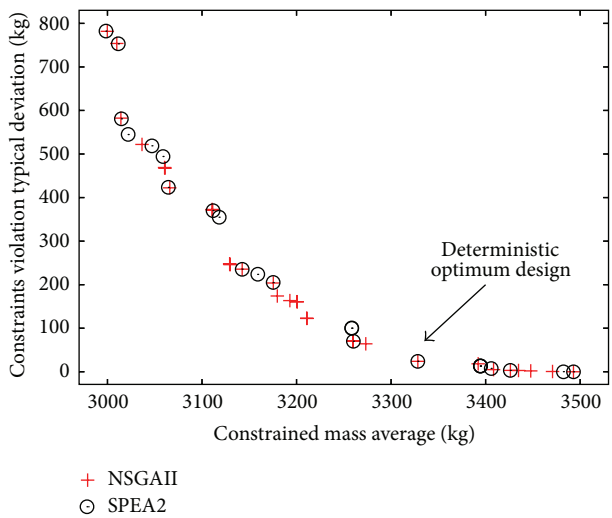


Figure 5: Final nondominated front obtained by accumulated SPEA2 and NSGAII executions (constrained weight versus constraint violation typical deviation, kg). Reduced Distribution.

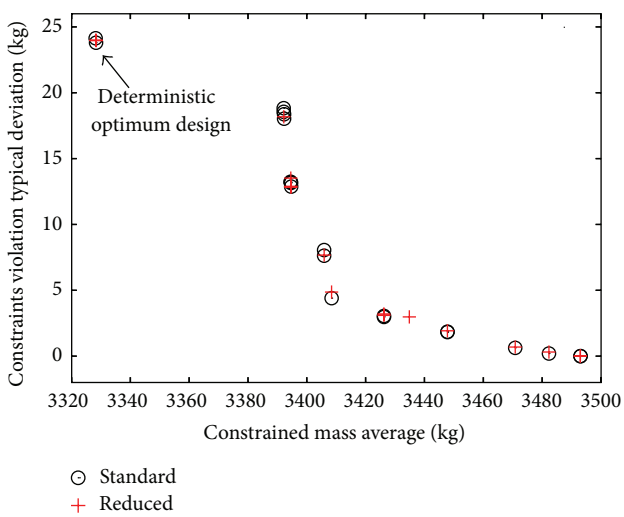


Figure 6: Final nondominated fronts obtained by the standard (circles) and reduced (crosses) distributions, starting with the deterministic optimum solution at top left. The two coincide (constrained weight versus constraint violation typical deviation, kg).

SPEA2 results are compared. Behaviour in the two cases is similar and they are both capable of locating the extreme frame structural design solutions. The number of final Pareto front designs located is similar to that of the nondominated front obtained by accumulating the whole solution set: 33 in NSGAII and 23 in SPEA2. Both algorithms also found all thirty-three design solutions, although the constraint variations caused by the Monte-Carlo simulation produce the effect that some of these points are dominated and not shown in the final front (Figures 5 and 6).

A comparison of these results with the standard procedure ones (as in [32, 43]) shows that the reduced simulation procedure obtains a wider front, both in terms of the number of structural designs obtained (33 versus 12) and in terms of numerical values (the left nondominated solutions reach 782.2 kg and 2998.4 kg in terms of typical deviation and constrained weight average, resp., versus the standard distribution values located furthest to the left of 79.3 kg and 3276.9 kg, resp.). This is due to the fact that the contribution to the constraint distribution in the reduced procedure is limited to the cases where the characteristic load value is exceeded, and therefore, there are structures with reduced lesser constraint average, which is added to the structural weight and considered in the x-axis fitness function value, than in the standard simulation where these structures were dominated designs (Figure 3) and so appear as new nondominated designs in the nondominated set (Figure 5).

It is worth pointing out that if the engineer or decision maker's preferences are properly considered, both procedures (the standard Monte-Carlo and the reduced procedure) achieve equivalent nondominated frame designs. The engineer or decision maker has to choose a structural design between two extremes of the nondominated frontier:

(1) the extreme right, where the solution of the nondominated set (highest constrained weight and null typical deviation) that represents the structural design has no constraint violation, even when considering uncertainties;

(2) the extreme left, where the solution of the nondominated set in the case in which the uncertainties of the load were not considered has the lowest weight and no constraint violation, which corresponds to the structural design that coincides with the deterministic optimum.

Therefore, when the useful functional space is restricted to the above-mentioned extreme points, both procedures (standard and reduced) lead to the same nondominated design set. This can be seen in Figure 6, where the accumulated nondominated fronts of both the SPEA2 and NSGAII algorithms and of each procedure (the standard and the reduced) are depicted. This, in turn, is explained because the two simulation procedures produce identical constraint distributions in those nondominated structural designs, where the characteristic load values are surpassed. Indeed, the first design where the two procedures coincide becomes the deterministic optimum design solution; so, a comparison of the two distributions could lead to a method that achieves the deterministic optimum design, where no load uncertainties are involved. The slight variations in the depicted solutions are due to the stochastic nature of the Monte-Carlo simulation, but the proximate points represent the same structural design.

In Figure 7, the whole set of accumulated nondominated solutions are shown graphically, differentiating the evolutionary multiobjective algorithms with which they were obtained. These designs are detailed numerically in Table 1, where the bold type corresponds to the deterministic optimum design. This table shows the constrained weight (x-axis fitness function, obtained by adding the structural weight and the constraint violation distribution average), constraint

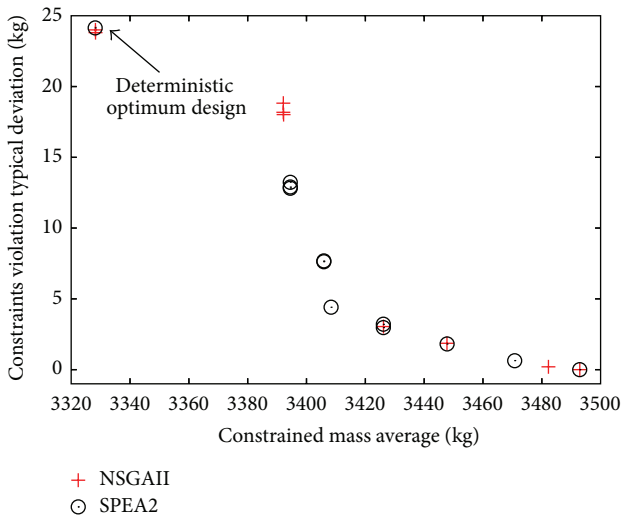


Figure 7: Final accumulated nondominated front obtained by accumulated standard and reduced distributions (constrained weight versus constraint violation typical deviation, kg).

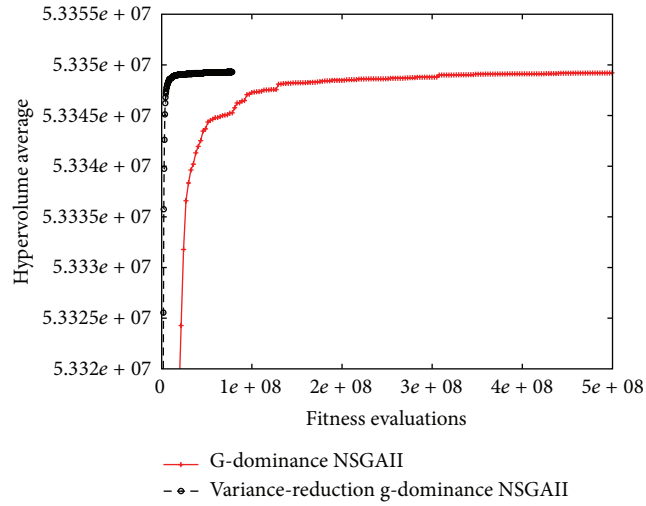


Figure 8: Average of S-metric (30 independent executions), NSGAII with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 100.

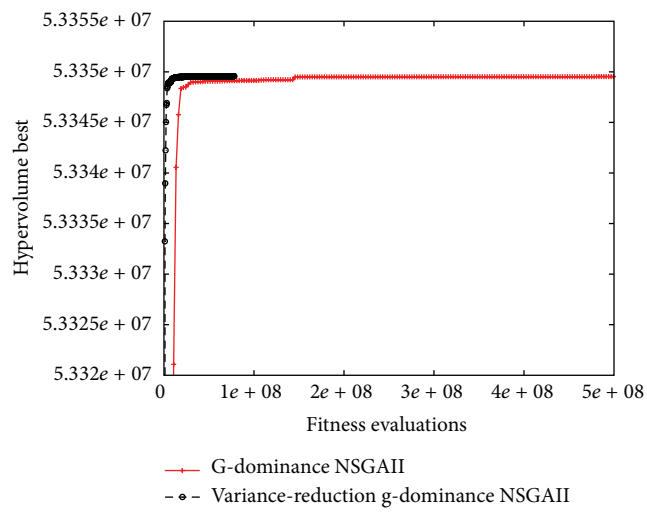


Figure 9: Best of S-metric (30 independent executions), NSGAII with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 100.

violation typical deviation (y-axis fitness function), constraint violation average, and detailed cross-section, as well as the algorithm type, for each nondominated structural design corresponding to those preferences that are useful for the decision maker. A total of ten structural designs were identified.

5.3. S-Metric Statistics. The reference point in the g-dominance is (3324.3, 0), the first value being that which corresponds to the minimum constrained weight of the deterministic optimum design (as published in [41]) and the second value corresponding to the aspiration level of the standard deviation of the violation of constraints distribution, which is a null variation.

In addition, in order to compare the convergence of the algorithm, the S-metric (or hypervolume metric), originally proposed by Zitzler and Thiele [45, 46], has been calculated as "running metrics" [47]. This is an unary metric that evaluates both approximation and coverage of the nondominated front. Specifically, the S-metric evaluation proposal of Fonseca et al. [48] is considered. The reference point for its calculation is (30000, 2000), a value which guarantees that it is dominated by the design solutions. Results are shown in Figures 8 to 19, where the horizontal axis represents the number of structural finite element evaluations and the y-axis represents the values of S-metric average (Figures 8, 11, 14, and 17), best (Figures 9, 12, 15, and 18) and standard deviation (Figures 10, 13, 16, and 19), out of thirty independent executions of NSGAII (Figures 8 to 13) and SPEA2 (Figures 14 to 19). Two population sizes are shown (100 and 200 individuals) for both NSGAII and SPEA2 algorithms, as are the two simulation procedures (standard and reduced).

All the Figures 8 to 19 show clearly the significant advantage of the combination of g-dominance with engineering knowledge-based variance-reduction simulation (black dots) versus the standard Monte-Carlo simulation (red crosses), in terms of better averages, best and standard deviation values in all the algorithms (NSGAII and SPEA2), and both population sizes (100 and 200).

5.4. An Additional Test Case: Increasing Loads. In this second test case, as explained at the end of Section 4, we have considered an increment of 5% in the loads; therefore, in the deterministic case, each load is increased by 5% in value; in the case of uncertainty handling, each of the three independent Gaussian distributions is increased by 5% in mean value and their standard deviations remain constant. The optimization results are obtained, performing 10 independent executions, a population size of 200, a stop criterion of 100 generations, a mutation rate of 6%, and uniform crossover.

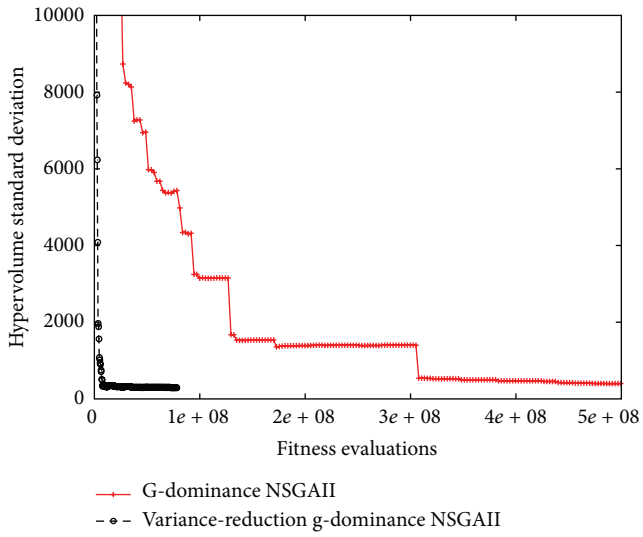


Figure 10: Standard deviation of S-metric (30 independent executions), NSGAII with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 100.

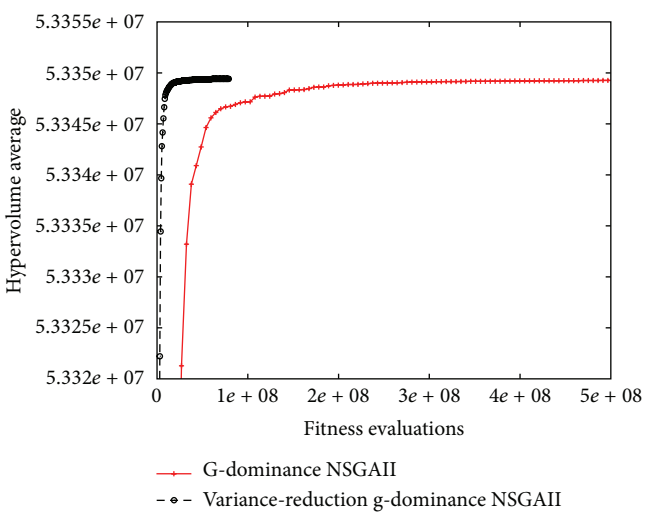


Figure 11: Average of S-metric (30 independent executions), NSGAII with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 200.

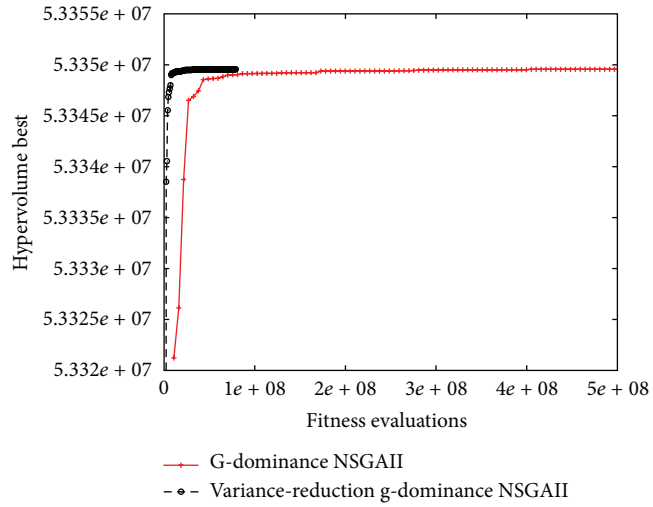


Figure 12: Best of S-metric (30 independent executions), NSGAII with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 200.

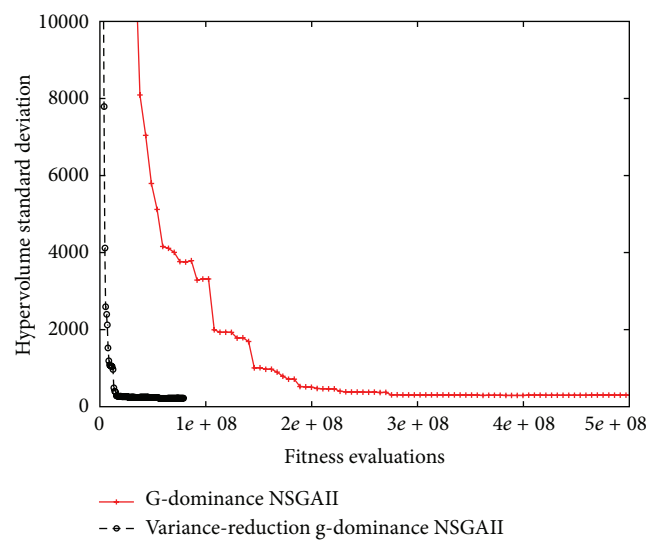


Figure 13: Standard deviation of S-metric (30 independent executions), NSGAII with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 200.

After optimization, the deterministic optimum design gives a weight of 3408.16 kg and its constraint violation is null (the design cross section types are Bar 1 = IPE-400, Bar 2 = IPE-550, Bar 3 = IPE-160, and Bar 4 = IPE-500).

Robust design results are shown in Table 2 and represented graphically in Figure 20. The design furthest to the left belonging to the nondominated front is represented in Table 2 in bold type, and it corresponds again to the deterministic optimum design (whose fitness function values have been considered in the reference point of the g-dominance). A total of eleven different designs make up the nondominated solutions. Both algorithms (NSGAII and SPEA2) are once again able to obtain the whole front; the best solutions of accumulated fronts are shown with an indication of which algorithm achieved each.

Figure 21 shows a comparison between the best designs of the original test case (represented by crosses) and the incremented loading test case (represented by circles). In the increased load case, the whole front is displaced to the right, which constitutes an increment in the optimum design weight when the load values are increased. A comparison with the solution furthest to the right (which has null constraint violations) in Tables 1 and 2 shows an increment of 5% in the value of the loads corresponding to an increment of 2.01% in the weight (3563.27 kg in Table 2 versus 3492.94 kg

Table 2: Detailed nondominated structural frame designs of Figure 20 (5% increased loads).

Constrained weight (kg)	Constraint violation typical deviation (kg)	Constraint violation average (kg)	Cross section type bar 1	Cross section type bar 2	Cross section type bar 3	Cross section type bar 4	Obtained with the EMO algorithm
3411.26	**22.05**	**3.101**	**IPE400**	**IPE550**	**IPE160**	**IPE500**	NSGAII
3428.14	18.274	2.083	IPE400	IPE550	IPE180	IPE500	SPEA2
3448.96	13.314	1.238	IPE400	IPE550	IPE200	IPE500	NSGAII
3471.49	8.943	0.696	IPE400	IPE550	IPE220	IPE500	SPEA2
3482.62	6.745	0.356	IPE360	IPE550	IPE360	IPE450	SPEA2
3493.04	3.204	0.103	IPE400	IPE550	IPE160	IPE550	SPEA2
3493.04	3.096	0.103	,,	,,	,,	,,	NSGAII
3510.88	1.696	0.042	IPE400	IPE550	IPE180	IPE550	SPEA2
3532.51	0.938	0.013	IPE400	IPE550	IPE200	IPE550	SPEA2
3532.51	0.717	0.013	,,	,,	,,	,,	NSGAII
3537.84	0.526	0.007	IPE360	IPE550	IPE400	IPE450	NSGAII
3537.84	0.505	0.007	,,	,,	,,	,,	SPEA2
3555.58	0.157	0.001	IPE400	IPE550	IPE220	IPE550	NSGAII
3563.27	0.0	0.0	IPE360	IPE550	IPE360	IPE500	NSGAII

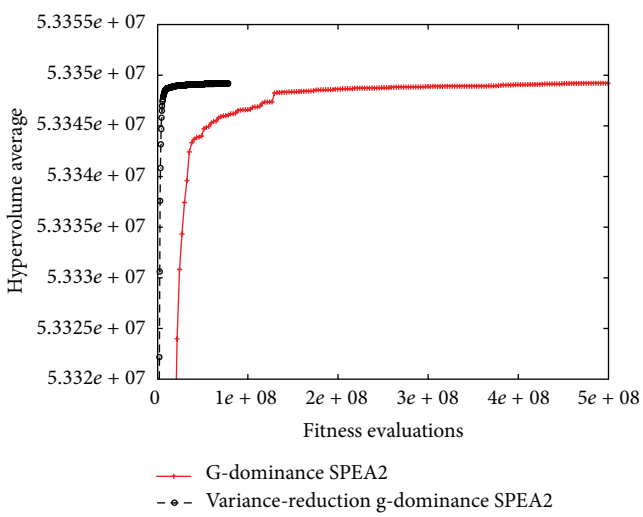


Figure 14: Average of S-metric (30 independent executions), SPEA2 with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 100.

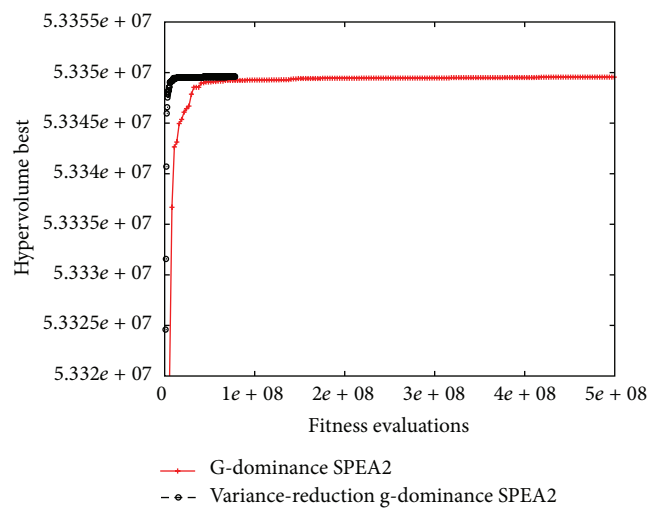


Figure 15: Best of S-metric (30 independent executions), SPEA2 with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 100.

in Table 1). Moreover, some designs belong to both nondominated fronts; specifically, the six lowest designs of Table 1 overlap with the six highest designs of Table 2, as depicted with letters A, B, C, D, E, and F in Figure 21 (each letter corresponds to the same structural design).

6. Conclusions

This paper has proposed, implemented, and verified a procedure for the robust optimum design of frame structures with real discrete cross-section types consisting of the simultaneous incorporation of engineering knowledge through:

(a) an engineering knowledge-based variance-reduction Monte-Carlo simulation based on important sampling methods, and

(b) the incorporation of the engineer/decision maker's preferences, as desirable aspiration levels, in terms of the consideration of the deterministic optimum design values (knowledge of variable values is not required) as a reference point in the g-dominance implemented in evolutionary multiobjective algorithms. This second aspect could be applied with no loss of generality with any other reference point preference-based evolutionary algorithm.

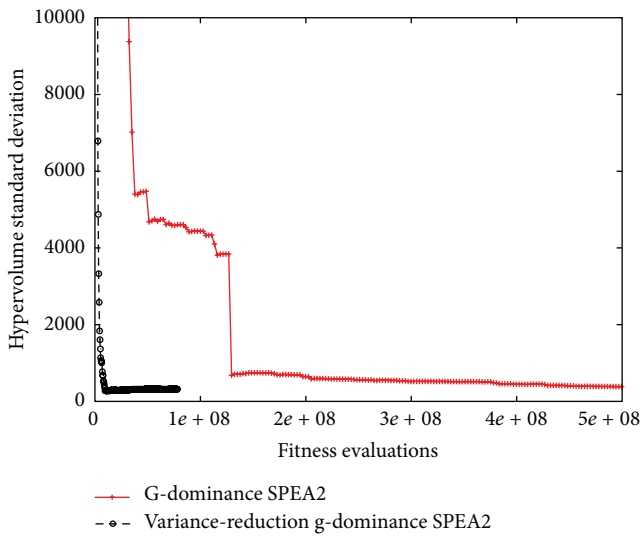


Figure 16: Standard deviation of S-metric (30 independent executions), SPEA2 with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 100.

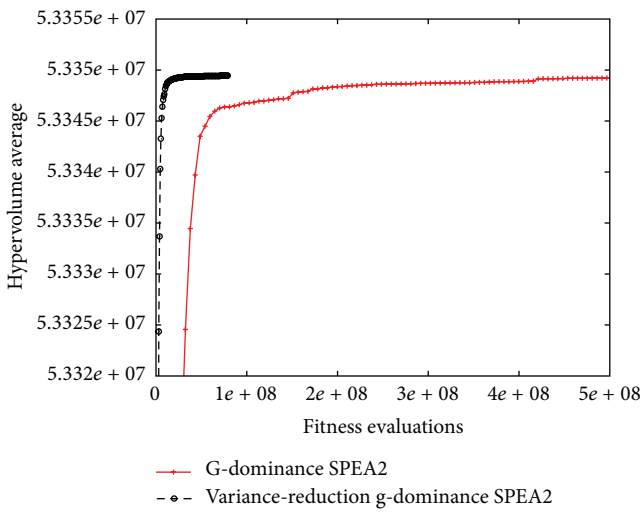


Figure 17: Average of S-metric (30 independent executions), SPEA2 with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 200.

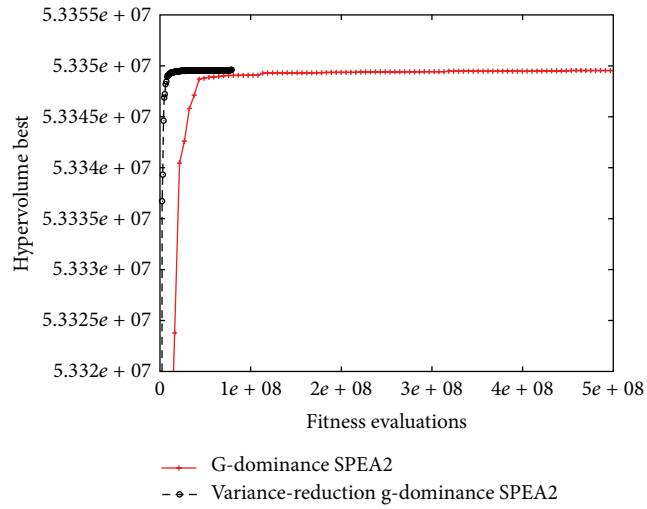


Figure 18: Best of S-metric (30 independent executions), SPEA2 with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 200.

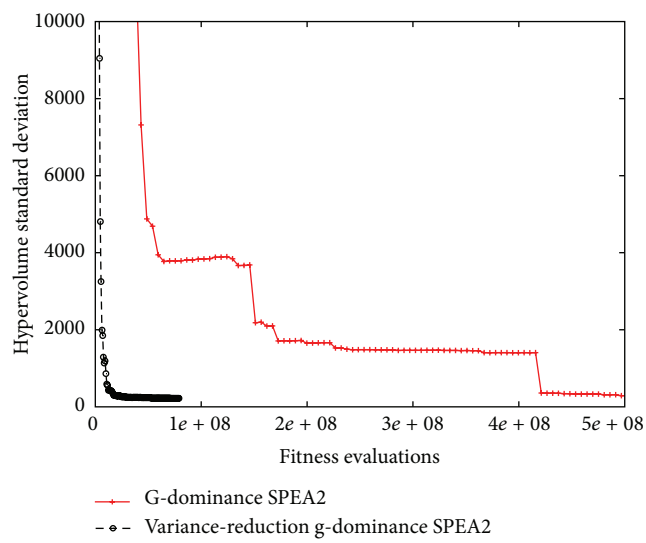


Figure 19: Standard deviation of S-metric (30 independent executions), SPEA2 with g-dominance, comparing standard and variance reduction Monte-Carlo. Population size is 200.

Uncertainty has been dealt with by including all the possible factors of constraints violation (stresses, displacements, and buckling effect) in the final designs. Two objectives were simultaneously minimized: first, the constrained weight (representative of system performance), by adding the weight of the structure and the average constraint violation penalty distribution; second, the typical deviation of the constraints violation penalty distribution (representative of system sensitivity).

Although each of these two engineering knowledge aspects alone is useful per se, the combination of the two adds considerable value as shown in the statistical analysis of the S-metric in terms of average, best, and standard deviation in the frame structural test case addressed. This methodology enhances efficiency in the global stochastic search in robust optimum structural engineering design.

Acknowledgments

This work has been partially funded by Research Project ULPGC2008-009. Also computational resources are supported by Project UNLP08-3E-2010 of the Secretaría de

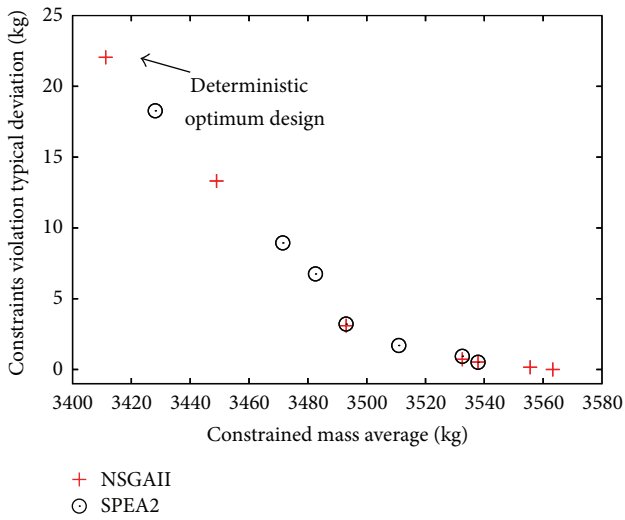


Figure 20: Final nondominated front obtained by accumulated SPEA2 and NSGAII (constrained weight versus constraint violation typical deviation). Reduced distribution (5% increased loads).

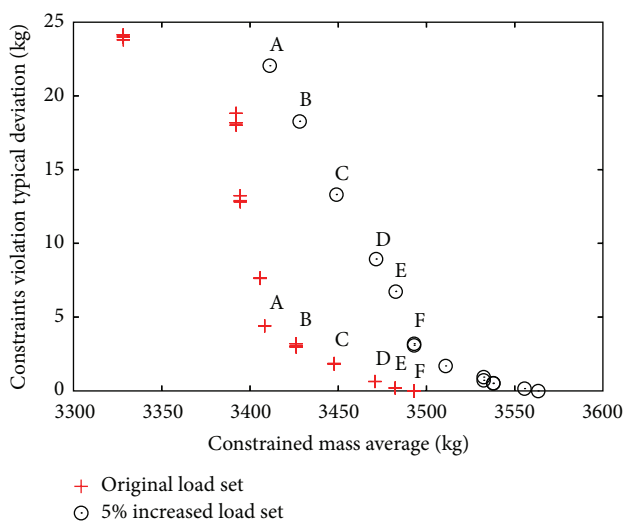


Figure 21: Final nondominated front obtained by accumulated SPEA2 and NSGAII (constrained weight versus constraint violation typical deviation). Comparing test cases.

Estado de Universidades e Investigación, Ministerio de Ciencia e Innovación (Spain), and FEDER.

References

[1] D. Goldberg and M. Samtani, "Engineering optimization via genetic algorithm," in *Proceedings of the 9th Conference on Electronic Computation (ASCE '86)*, pp. 471–482, New York, NY, USA, 1986.

[2] "Recent advances in optimal structural design," S. A. Burns, Ed., Institute of American Society of ASCE-SEI, 2002.

[3] R. Kicinger, T. Arciszewski, and K. A. de Jong, "Evolutionary computation and structural design: a survey of the state-of-the-art," *Computers and Structures*, vol. 83, no. 23-24, pp. 1943–1978, 2005.

[4] D. Greiner, J. M. Emperador, B. Galván, G. Winter, and J. Periaux, "Optimum structural design using bio-inspired search methods: a survey and applications," in *Computational Intelligence in Aerospace Sciences*, V. Becerra and M. Vasile, Eds., American Institute of Aeronautics and Astronautics (AIAA), 2014.

[5] A. H. Gandomi, X. S. Yang, S. Talahari, and A. H. Alavi, *Metaheuristic Applications in Structures and Infrastructures*, Elsevier, 1st edition, 2013.

[6] C. A. Coello Coello, "Evolutionary multi-objective optimization: a historical view of the field," *IEEE Computational Intelligence Magazine*, vol. 1, no. 1, pp. 28–36, 2006.

[7] A. K. Dhingra and B. H. Lee, "A genetic algorithm approach to single and multi-objective structural optimization with discrete-continuous variables," *International Journal for Numerical Methods in Engineering*, vol. 37, pp. 4059–4080, 1994.

[8] C. A. Coello and A. D. Christiansen, "Multi-objective optimization of trusses using genetic algorithms," *Computers & Structures*, vol. 75, no. 6, pp. 647–660, 2000.

[9] D. Greiner, J. M. Emperador, and G. Winter, "Single and Multi-objective frame optimization by evolutionary algorithms and the auto-adaptive rebirth operator," *Computer Methods in Applied Mechanics and Engineering*, vol. 193, no. 33–35, pp. 3711–3743, 2004.

[10] L. Rachmawati and D. Srinivasan, "Preference incorporation in multi-objective evolutionary algorithms: a survey," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 962–968, British Columbia, Canada, July 2006.

[11] J. Branke and K. Deb, "Integrating user preferences into evolutionary multi-objective optimization," in *Knowledge Incorporation in Evolutionary Computation*, Y. Jin, Ed., vol. 167 of *Studies in Fuzziness and Soft Computing*, pp. 461–478, Springer, 2005.

[12] D. Cvetkovic and C. Coello, "Human preferences and their applications in evolutionary multi-objective optimization," in *Knowledge Incorporation in Evolutionary Computation*, Y. Jin, Ed., vol. 167 of *Studies in Fuzziness and Soft Computing*, pp. 479–502, Springer, 2005.

[13] J. Wang and J. P. Terpenny, "Interactive preference incorporation in evolutionary engineering design," in *Knowledge Incorporation in Evolutionary Computation*, Y. Jin, Ed., vol. 167 of *Studies in Fuzziness and Soft Computing*, pp. 525–543, Springer, 2005.

[14] L. Ben Said, S. Bechikh, and K. Ghedira, "The r-dominance: a new dominance relation for interactive evolutionary multi-criteria decision making," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 5, pp. 801–818, 2010.

[15] K. Deb, J. Sundar, R. Bhaskara, and S. Chaudhuri, "Reference point based multi-objective optimization using evolutionary algorithms," *International Journal of Computational Intelligence Research*, vol. 2-3, pp. 273–286, 2006.

[16] L. Thiele, K. Miettinen, P. J. Korhonen, and J. Molina, "A preference-based evolutionary algorithm for multi-objective optimization," *Evolutionary Computation*, vol. 17, no. 3, pp. 411–436, 2009.

[17] J. Molina, L. V. Santana, A. G. Hernández-Díaz, C. A. Coello Coello, and R. Caballero, "G-dominance: reference point based dominance for Multi-objective metaheuristics," *European Journal of Operational Research*, vol. 197, no. 2, pp. 685–692, 2009.

[18] P. Breitkopf and R. F. Coelho, *Design Optimization in Computational Mechanics*, John Wiley & Sons, 2010.

[19] Y. Tsompanakis, N. D. Lagaros, and M. Papadrakakis, "Structural design optimization considering uncertainties," in *Structures and Infrastructures Series*, D. Frangopol, Ed., vol. 1, Taylor & Francis, Boca Raton, Fla, USA, 2008.

[20] H. G. Beyer and B. Sendhoff, "Robust optimization—a comprehensive survey," *Computer Methods in Applied Mechanics and Engineering*, vol. 196, no. 33-34, pp. 3190–3218, 2007.

[21] G. I. Schuëller and H. A. Jensen, "Computational methods in optimization considering uncertainties—an overview," *Computer Methods in Applied Mechanics and Engineering*, vol. 198, no. 1, pp. 2–13, 2008.

[22] R. F. Coelho, "Co-evolutionary optimization for multi-objective design under uncertainty," *Journal of Mechanical Design ASME*, vol. 135, no. 2, pp. 1–8, 2013.

[23] D. Greiner and P. Hajela, "Truss topology optimization for mass and reliability considerations-co-evolutionary Multi-objective formulations," *Structural and Multidisciplinary Optimization*, vol. 45, pp. 589–613, 2012.

[24] G. J. Park, T. H. Lee, K. H. Lee, and K. H. Hwang, "Robust design: an overview," *AIAA Journal*, vol. 44, no. 1, pp. 181–191, 2006.

[25] I. Doltsinis, Z. Kang, and G. Cheng, "Robust design of nonlinear structures using optimization methods," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 12–16, pp. 1779–1795, 2005.

[26] E. Sandgren and T. M. Cameron, "Robust design optimization of structures through consideration of variation," *Computers and Structures*, vol. 80, no. 20-21, pp. 1605–1613, 2002.

[27] N. Lagaros, V. Plevris, Y. Tsompanakis, and M. Papadrakakis, "Multi-performance robust optimum design of steel structures," in *Proceedings of the 6th World Congress of Structural and Multidisciplinary Optimization*, 2005.

[28] N. D. Lagaros, V. Plevris, and M. Papadrakakis, "Multi-objective design optimization using cascade evolutionary computations," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 30–33, pp. 3496–3515, 2005.

[29] N. D. Lagaros and M. Papadrakakis, "Robust seismic design optimization of steel structures," *Structural and Multidisciplinary Optimization*, vol. 37, no. 6, pp. 457–469, 2007.

[30] N. D. Lagaros, Y. Tsompanakis, M. Fragiadakis, V. Plevris, and M. Papadrakakis, "Metamodel-based computational techniques for solving structural optimization problems considering uncertainties," in *Structural Design Optimization Considering Uncertainties (Structures and Infrastructures Series)*, Tsompanakis, Lagaros, and Papadrakakis, Eds., Taylor & Francis, 2008.

[31] C. Zang, M. I. Friswell, and J. E. Mottershead, "A review of robust optimal design and its application in dynamics," *Computers and Structures*, vol. 83, no. 4-5, pp. 315–326, 2005.

[32] D. Greiner, B. Galvan, J. M. Emperador, and G. Winter, "Structural robust design optimization of steel frames with engineering knowledge-based variance-reduction simulation," in *Proceedings of the European Conference of Civil Engineering (ECCIE '10)*, pp. 13–18, 2010.

[33] D. Greiner, B. Galván, J. M. Emperador, M. Méndez, and G. Winter, "Introducing reference point using g-dominance in optimum design considering uncertainties: an application in structural engineering," in *Evolutionary Multi-Criterion Optimization*, vol. 6576 of *Lecture Notes in Computer Science*, pp. 389–403, 2011.

[34] ISO, "General principles on reliability for structures," International Organization for Standardization (ISO), no. 2394, 1998.

[35] J. Schneider, "Introduction to safety and reliability of structures," Structural Engineering Documents 5, IABSE, Zurich, Switzerland, 1997.

[36] CEB, *Reliability of Concrete Structures. Final Report of Commission I*, vol. 202 of *CEB Bulletin no. 112*, 1991.

[37] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist Multi-objective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[38] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: improving the strength pareto evolutionary algorithm for multi-objective optimization," in *Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems (Eurogen)*, K. C. Giannakoglou, D. T. Tsahalis, J. Periaux, K. D. Papailiou, and T. Fogarty, Eds., pp. 95–100, John Wiley & Sons, 2001.

[39] S. Hernández Ibáñez, "Structural optimal design methods," Seinor Collection, Col. Ingenieros Caminos, Canales y Puertos (Madrid), 1990, (Spanish).

[40] D. Greiner, G. Winter, and J. M. Emperador, "Optimising frame structures by different strategies of genetic algorithms," *Finite Elements in Analysis and Design*, vol. 37, no. 5, pp. 381–402, 2001.

[41] D. Greiner, J. M. Emperador, and G. Winter, "Multi-objective optimization of bar structures," in *Proceedings of the European Congress on Computational Methods in Applied Sciences and Engineering*, G. A. Pareto, Ed., CIMNE, Barcelona, Spain, 2000.

[42] D. Greiner, G. Winter, and J. M. Emperador, "A comparative study about the mutation rate in multi-objective frame structural optimization using evolutionary algorithms," in *Proceedings of the 6th Conference on Evolutionary and Deterministic Methods forDesign, Optimization and Control with Applications to Industrial and Societal Problems (EUROGEN '05)*, R. Schilling, W. Haase, J. Periaux, and H. Baier, Eds., Munich, Germany, 2005.

[43] D. Greiner, J. M. Emperador, B. Galván, and G. Winter, "Robust design of frames under uncertain loads by multi-objective genetic algorithms," in *Proceedings of the 8th International Conference on Computational Structures Technology*, B. Topping, G. Montero, and R. Montenegro, Eds., vol. 200, Civil-Comp-Press, Stirlingshire, UK, 2006.

[44] G. S. Fishman, *Monte-Carlo: Concepts, Algorithms and Applications*, Springer Series in Operations Research, Springer, New York, NY, USA, 1996.

[45] E. Zitzler and L. Thiele, "Multi-objective optimization using evolutionary algorithms—a comparative case study," in *Parallel Problem Solving From Nature*, A. E. Eiben, T. Bäck, M. Schoenauer, and H. P. Schwefel, Eds., vol. 1498 of *Lecture Notes in Computer Science*, pp. 292–301, 1998.

[46] E. Zitzler and L. Thiele, "Multi-objective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

[47] K. Deb and S. Jain, "Running performance metrics for evolutionary multi-objective optimizations," in *Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning (SEAL '02)*, pp. 13–20, 2002.

[48] C. M. Fonseca, L. Paquete, and M. López-Ibáñez, "An improved dimension-sweep algorithm for the hypervolume indicator," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 1157–1163, Vancouver, Canada, July 2006.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 589510, 11 pages
http://dx.doi.org/10.1155/2013/589510

Research Article

Intelligent Motion Control for Four-Wheeled Holonomic Mobile Robots Using FPGA-Based Artificial Immune System Algorithm

Hsu-Chih Huang

Department of Electrical Engineering, National Ilan University, No. 1, Section 1, Shen-Lung Road, Yilan City 26047, Taiwan

Correspondence should be addressed to Hsu-Chih Huang; hchuang@niu.edu.tw

Received 8 August 2013; Revised 17 October 2013; Accepted 17 October 2013

Academic Editor: Amir H. Alavi

This paper presents an intelligent motion controller for four-wheeled holonomic mobile robots with four driving omnidirectional wheels equally spaced at 90 degrees from one another by using field-programmable gate array (FPGA)-based artificial immune system (AIS) algorithm. Both the nature-inspired AIS computational approach and motion controller are implemented in one FPGA chip to address the optimal control problem of real-world mobile robotics application. The proposed FPGA-based AIS method takes the advantages of artificial intelligence and FPGA technology by using system-on-a-programmable chip (SoPC) methodology. Experimental results are conducted to show the effectiveness and merit of the proposed FPGA-based AIS intelligent motion controller for four-wheeled omnidirectional mobile robots. This FPGA-based AIS autotuning intelligent controller outperforms the conventional nonoptimal controllers, the genetic algorithm (GA) controller, and the particle swarm optimization (PSO) controller.

1. Introduction

In the last few years, holonomic mobile robots with omnidirectional driving capability have attracted much attention in both mechanical engineering and electrical engineering. Such robots are superior to those with differential wheels in terms of dexterity and locomotion mechanism [1–5]. Comparing with the car-like robots [6–8], this type of omnidirectional mobile mechanism takes the advantages in maneuverability and effectiveness to move in any direction at any time [1–5].

Modeling and control of holonomic four-wheeled omnidirectional mobile robots have been investigated by several researchers [9–13]. In particular, Byun et al. [10] constructed an omnidirectional mobile robot with a variable wheel arrangement mechanism. Purwin and D'Andrea [11] proposed the trajectory generation for four wheeled omnidirectional vehicles. Shing et al. [12] proposed a T-S fuzzy path controller design for an omnidirectional mobile robot. The dynamic model and fuzzy controller for an omnidirectional surveillance robot were presented in [13]. Moreover, Tsai et al. [14] presented a dynamic model to achieve adaptive motion control. However, none of the above studies coped with the control parameter optimization problems, namely, that the control parameters in their proposed controllers are poorly tuned or determined by an ineffective trial-and-error approach.

Artificial intelligence (AI) is a set of nature-inspired computational methodologies for solving the optimization problems in a wide variety of real-world applications [15–26]. Among these modern metaheuristic-based approaches to address the optimization problems in mobile robots, AIS algorithm inspired by the biological immune system is introduced by de Castro and Timmis [15] and has emerged as a highly effective and efficient computational paradigm to NP-hard combinatorial optimization problems. Its model is directly related to the natural immune response. The biological immune system is a complex system of specialized cells that has evolved to protect the body against attacks from outside invaders such as bacteria, viruses, fungi, and parasites in biological world [15–20]. The specific immune cells, called antibodies, are produced to recognize the foreign antigens and then kill them in the adaptive immune response. The antibodies (optimal solution) with highest affinity are evolved via the AIS algorithm to solve the optimization problems (antigens).

The adaptive and robust AIS algorithms have been successfully applied in many disciplines by exploiting their strong optimization ability [15–20], such as machine learning, classification, data analysis, optimization, and pattern recognition. Zhang and Xin [16] presented an immune algorithm with adaptive sampling in noisy environments and its application to stochastic optimization problems. Chen and You [17] proposed an immune algorithms-based approach for redundant reliability problems with multiple component choices. Tsai and Chou [18] presented the design of optimal digital IIR filters by using an improved immune algorithm. Lin and Chen [19] presented the parameter tuning, feature selection, and weight assignment of features for case-based reasoning by artificial immune system. Kim [20] developed an intelligent tuning method of 2-DOF (degree of freedom) proportional-integral-derivative (PID) controller by immune algorithm. Unfortunately, these studies did not address the AIS-based intelligent motion control problem of holonomic mobile robots. Moreover, these personal computer (PC)-based and workstation-based AISs suffer from their high cost and software implementation so that they are not suitable for real-time robotics applications. This disadvantage of the PC-based algorithms can be circumvented by using the SoPC hardware/software codesign technology in FPGA chip [27–30].

FPGA technology has been bringing a major revolution in the design of integrated circuits and has been emerging as a new paradigm for AI computing. It has been shown as an efficient and powerful means to realize the sophisticated algorithms in many disciplines, such as fuzzy controller design [27], speech recognition [28], neural network controller design [29], and positioning algorithm [30]. With the benefits of flexibility, hardware/software codesign, and IP (intellectual property) reusability, this FPGA computing has been shown useful to design computational intelligence in embedded applications which contains memory, custom logics, and processor cores [27–30]. However, there has been no attempt to designing an intelligent FPGA-based AIS motion controller for holonomic mobile robots.

The objective of this paper is to design a pragmatic FPGA-based intelligent AIS motion controller for four-wheeled omnidirectional mobile robots to achieve both trajectory tracking and stabilization. With the kinematic model of the omnidirectional mobile robot, the unified control law is proposed. Moreover, the control parameters are then optimized by using the proposed AIS algorithm, thereby improving the control performance. Finally, the AIS computational approach is realized in FPGA to efficiently construct an experimental holonomic four-wheeled mobile robot.

The rest of this paper is organized as follows. In Section 2, the kinematic control law is proposed to achieve stabilization and trajectory tracking for the four-wheeled holonomic mobile robots. Section 3 elaborates the AIS algorithm and its application to control parameter tuning. Section 4 presents the FPGA implementation of the AIS computational paradigm to motion controller. Section 5 conducts experimental results to show the performance and merit of the proposed methods. Section 6 concludes this paper.

2. Kinematic Control

This section is devoted to briefly describing the kinematic model of a holonomic omnidirectional mobile robot with four independent driving wheels equally spaced at 90 degrees from one another. With the kinematic model, the unified kinematic controller is proposed to achieve stabilization and trajectory tracking.

2.1. Kinematic Model. Figure 1 depicts the structure and geometry of the four-wheeled holonomic omnidirectional driving configuration with respect to a world frame. Due to structural symmetry, the platform has the property that the center of geometry coincides with the center of mass. What follows describes the kinematic model of this kind of robot, where θ represents the vehicle orientation which is positive in the counterclockwise direction. Note that θ also denotes the angle between the moving frame and the world frame. The kinematic model of the four-wheeled holonomic mobile robot is presented in [9–13]:

$$v(t) = \begin{bmatrix} v_1(t) \\ v_2(t) \\ v_3(t) \\ v_4(t) \end{bmatrix} = \begin{bmatrix} r\omega_1(t) \\ r\omega_2(t) \\ r\omega_3(t) \\ r\omega_4(t) \end{bmatrix} = P(\theta(t)) \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix}, \quad (1)$$

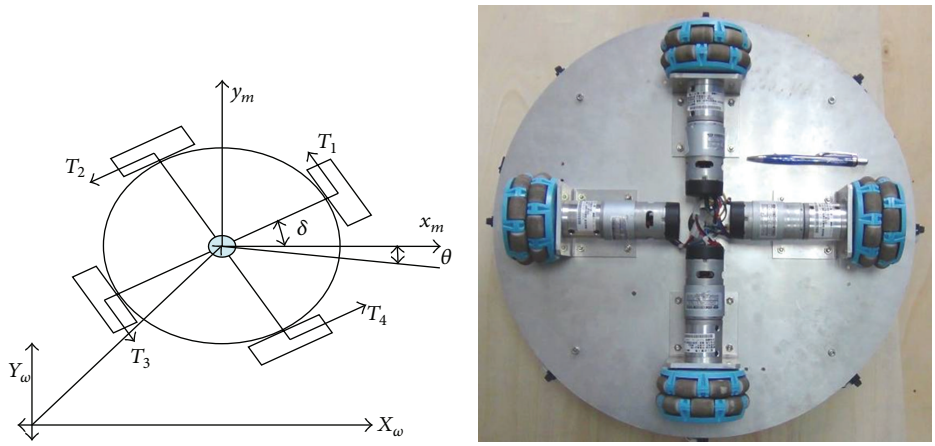
where

$$P(\theta(t)) = \begin{bmatrix} -\sin(\delta+\theta) & \cos(\delta+\theta) & L \\ -\cos(\delta+\theta) & -\sin(\delta+\theta) & L \\ \sin(\delta+\theta) & -\cos(\delta+\theta) & L \\ \cos(\delta+\theta) & \sin(\delta+\theta) & L \end{bmatrix}, \quad (2)$$

where r denotes the radius of the driving wheel; L represents the distance from the wheel's center to the center to the geometric center of the mobile platform; $v_i(t)$ and $\omega_i(t)$, $i = 1, 2, 3, 4$, respectively, denote the linear and angular velocities of each wheel; $[x(t) \quad y(t) \quad \theta(t)]^T$ represent the pose of the holonomic mobile robot. Note that although the matrix $P(\theta(t))$ is singular for any θ, its left inverse matrix can be found, that is, $P^{\#}(\theta(t))P(\theta(t)) = I$, and expressed by

$$P^{\#}(\theta(t)) = \begin{bmatrix} \frac{-\sin(\delta+\theta)}{2} & \frac{-\cos(\delta+\theta)}{2} & \frac{\sin(\delta+\theta)}{2} & \frac{\cos(\delta+\theta)}{2} \\ \frac{\cos(\delta+\theta)}{2} & \frac{-\sin(\delta+\theta)}{2} & \frac{-\cos(\delta+\theta)}{2} & \frac{\sin(\delta+\theta)}{2} \\ \frac{1}{4L} & \frac{1}{4L} & \frac{1}{4L} & \frac{1}{4L} \end{bmatrix}. \quad (3)$$

2.2. Kinematic Control. With the kinematic model in (1), this subsection is devoted to designing the kinematic controllers to achieve point stabilization and trajectory tracking for the holonomic omnidirectional mobile robot in Figure 1. Furthermore, the unified nonlinear control approach is also presented to simultaneously solve the stabilization and tracking problems in four-wheeled holonomic mobile robots.



(a) (b)

Figure 1: Structure and geometry of the four-wheeled holonomic omnidirectional mobile robot.

2.2.1. Point Stabilization. The control goal of the point stabilization is to find the controlled angular velocity vector $[\omega_1(t)\ \ \omega_2(t)\ \ \omega_3(t)\ \ \omega_4(t)]^T$ to steer the holonomic mobile robot from any starting pose $[x_0\ \ y_0\ \ \theta_0]^T$ to any desired destination pose $Z_d = [x_d\ \ y_d\ \ \theta_d]^T$. Note that the current pose of the mobile robot is $Z_1(t) = [x(t)\ \ y(t)\ \ \theta(t)]^T$. To design the motion controller, one defines the pose error which is the difference between the present pose and the desired destination pose; that is,

$$Z_e(t) = \begin{bmatrix} x_e(t) \\ y_e(t) \\ \theta_e(t) \end{bmatrix} = \begin{bmatrix} x(t) \\ y(t) \\ \theta(t) \end{bmatrix} - \begin{bmatrix} x_d \\ y_d \\ \theta_d \end{bmatrix} \tag{4}$$

which gives

$$\begin{bmatrix} \dot{x}_e(t) \\ \dot{y}_e(t) \\ \dot{\theta}_e(t) \end{bmatrix} = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = P^{\#}(\theta(t)) \begin{bmatrix} r\omega_1(t) \\ r\omega_2(t) \\ r\omega_3(t) \\ r\omega_4(t) \end{bmatrix}. \tag{5}$$

To asymptotically stabilize the control system, the following stabilization law is proposed. Note that the matrices K_P and K_I are symmetric and positive definite; that is, $K_P = \mathrm{diag}\{K_{p1}, K_{p2}, K_{p3}\} = K_P^T > 0$, $K_I = \mathrm{diag}\{K_{i1}, K_{i2}, K_{i3}\} = K_I^T > 0$:

$$\begin{bmatrix} \omega_1(t) \\ \omega_2(t) \\ \omega_3(t) \\ \omega_4(t) \end{bmatrix} = \frac{1}{r} P(\theta(t)) \times \left(-K_P \begin{bmatrix} x_e(t) \\ y_e(t) \\ \theta_e(t) \end{bmatrix} - K_I \begin{bmatrix} \int_0^t x_e(\tau)\,d\tau \\ \int_0^t y_e(\tau)\,d\tau \\ \int_0^t \theta_e(\tau)\,d\tau \end{bmatrix} \right). \tag{6}$$

The control system proposed in (6) takes the advantages of PID control theory which is widely used in industrial control systems [20]. Taking (6) into (5), the dynamics of the closed-loop error system becomes

$$\begin{bmatrix} \dot{x}_e(t) \\ \dot{y}_e(t) \\ \dot{\theta}_e(t) \end{bmatrix} = -K_P \begin{bmatrix} x_e(t) \\ y_e(t) \\ \theta_e(t) \end{bmatrix} - K_I \begin{bmatrix} \int_0^t x_e(\tau)\,d\tau \\ \int_0^t y_e(\tau)\,d\tau \\ \int_0^t \theta_e(\tau)\,d\tau \end{bmatrix}. \tag{7}$$

For the asymptotical stability of the closed-loop error system, a radially unbounded Lyapunov function candidate is chosen as follows:

$$\begin{aligned} V_1(t) = {} & \frac{1}{2} [x_e(t)\ \ y_e(t)\ \ \theta_e(t)] \begin{bmatrix} x_e(t) \\ y_e(t) \\ \theta_e(t) \end{bmatrix} \\ & + \frac{1}{2} \left[\int_0^t x_e(\tau)\,d\tau\ \ \int_0^t y_e(\tau)\,d\tau\ \ \int_0^t \theta_e(\tau)\,d\tau \right] K_I \\ & \times \begin{bmatrix} \int_0^t x_e(\tau)\,d\tau \\ \int_0^t y_e(\tau)\,d\tau \\ \int_0^t \theta_e(\tau)\,d\tau \end{bmatrix}. \end{aligned} \tag{8}$$

Taking the time derivative of $V_1(t)$, one obtains

$$\begin{aligned} \dot{V}_1(t) = {} & [x_e(t)\ \ y_e(t)\ \ \theta_e(t)] \begin{bmatrix} \dot{x}_e(t) \\ \dot{y}_e(t) \\ \dot{\theta}_e(t) \end{bmatrix} \\ & + \left[\int_0^t x_e(\tau)\,d\tau\ \ \int_0^t y_e(\tau)\,d\tau\ \ \int_0^t \theta_e(\tau)\,d\tau \right] K_I \end{aligned}$$

$$\times \begin{bmatrix} x_e(t) \\ y_e(t) \\ \theta_e(t) \end{bmatrix} = -\left[x_e(t) \;\; y_e(t) \;\; \theta_e(t)\right] K_P \begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} < 0. \tag{9}$$

Since $\dot{V}_1$ is negative semidefinite, Barbalat's lemma implies that $\left[x_e(t) \;\; y_e(t) \;\; \theta_e(t)\right]^T \to \left[0 \;\; 0 \;\; 0\right]^T$ as $t \to \infty$.

2.2.2. Trajectory Tracking. This subsection considers the trajectory tracking problem of the holonomic omnidirectional mobile robot. Unlike all nonholonomic conventional mobile robots, the trajectories of the omnidirectional mobile robots cannot be generated using their kinematic models; that is, any smooth and differentiable trajectories for the omnidirectional robots can be arbitrarily planned. Given the smooth and differentiable trajectory $Z_d(t) = \left[x_d(t) \;\; y_d(t) \;\; \theta_d(t)\right]^T \in C^1$, one defines the following tracking error vector:

$$Z_e(t) = \begin{bmatrix} x_e(t) \\ y_e(t) \\ \theta_e(t) \end{bmatrix} = \begin{bmatrix} x(t) \\ y(t) \\ \theta(t) \end{bmatrix} - \begin{bmatrix} x_d(t) \\ y_d(t) \\ \theta_d(t) \end{bmatrix}. \tag{10}$$

Thus, one obtains

$$\begin{aligned} \begin{bmatrix} \dot{x}_e(t) \\ \dot{y}_e(t) \\ \dot{\theta}_e(t) \end{bmatrix} &= \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} - \begin{bmatrix} \dot{x}_d(t) \\ \dot{y}_d(t) \\ \dot{\theta}_d(t) \end{bmatrix} \\ &= P^{\#}(\theta(t)) \begin{bmatrix} r\omega_1(t) \\ r\omega_2(t) \\ r\omega_3(t) \\ r\omega_4(t) \end{bmatrix} - \begin{bmatrix} \dot{x}_d(t) \\ \dot{y}_d(t) \\ \dot{\theta}_d(t) \end{bmatrix}. \end{aligned} \tag{11}$$

Similarly, the control goal is to find the motors' angular velocities $[\omega_1(t) \;\; \omega_2(t) \;\; \omega_3(t) \;\; \omega_4(t)]^T$ such that the closed-loop error system is globally asymptotical stable. In doing so, one proposes the following trajectory tracking law such that

$$\begin{aligned} &\begin{bmatrix} \omega_1(t) \\ \omega_2(t) \\ \omega_3(t) \\ \omega_4(t) \end{bmatrix} \\ &= \frac{1}{r} P(\theta(t)) \\ &\times \left(-K_P \begin{bmatrix} x_e(t) \\ y_e(t) \\ \theta_e(t) \end{bmatrix} - K_I \begin{bmatrix} \int_0^t x_e(\tau)\, d\tau \\ \int_0^t y_e(\tau)\, d\tau \\ \int_0^t \theta_e(\tau)\, d\tau \end{bmatrix} + \begin{bmatrix} \dot{x}_d(t) \\ \dot{y}_d(t) \\ \dot{\theta}_d(t) \end{bmatrix} \right), \end{aligned} \tag{12}$$

where the matrices, K_P and K_I, are symmetric and positive definite. Substituting (12) into (11) leads to the underlying closed-loop error system governed by

$$\begin{bmatrix} \dot{x}_e(t) \\ \dot{y}_e(t) \\ \dot{\theta}_e(t) \end{bmatrix} = -K_P \begin{bmatrix} x_e(t) \\ y_e(t) \\ \theta_e(t) \end{bmatrix} - K_I \begin{bmatrix} \int_0^t x_e(\tau)\, d\tau \\ \int_0^t y_e(\tau)\, d\tau \\ \int_0^t \theta_e(\tau)\, d\tau \end{bmatrix}. \tag{13}$$

Similar to point stabilization, the Lyapunov function candidate can be chosen as (8), and from (9) one can easily prove that the closed-loop error system for trajectory tracking control is asymptotically stable. Worthy of mention is that the point stabilization and trajectory tracking control problems can be simultaneously achieved by the control law (12). The unified control law (12) becomes a point stabilization one if the desired pose $\left[x_d(t) \;\; y_d(t) \;\; \theta_d(t)\right]^T$ can be either the time-dependent trajectory or the fixed destination posture.

3. AIS Algorithm for Intelligent Motion Control

AIS metaheuristic algorithms evolve improved solutions of the complex combinatorial problem by means of immune system, clonal selection, and hypermutation. Compared to GAs [21–23] and PSOs [24–26] bioinspired metaheuristics, this computational methodology with different evolutionary strategies is relatively new to solving real-world optimization problems in which conventional methodologies are ineffective or infeasible.

In order to evaluate the performance of the antibody, each antibody is assigned an affinity which is predefined in the optimization problem. The AIS antibodies with highest affinity are cloned, hypermutated, and selected, while random antibodies are also generated to improve the diversity of the population [15–20]. Worse antibodies are replaced by better mutated clones. This population evolves by the AIS process until the termination condition is met.

In what follows, the natural immune system is introduced and the core operations of the proposed AIS, such as antibody encoding scheme, affinity computation, clonal selection, and somatic hypermutation, are then presented. Finally, the proposed AIS computational paradigm is applied to design an intelligent motion controller for the holonomic mobile robot.

3.1. Natural Immune System. Biological immune system is a highly evolved, complicated adaptability system, which can identify and eliminate foreign antigenic pathogens to prevent damage to the body. An immune response is the body's reaction to antigens so that the antigens are eliminated [16–18]. The most important function of a biological immune system is to protect living organisms from invading antigens. There are two main types of lymphocytes: T cell and B cell. The T cell matures in the thymus and the B cell matures in the bone marrow. On the surface of T cells and B cells, there are receptor molecules that bind to others. These two cells are different in many aspects in immune system.

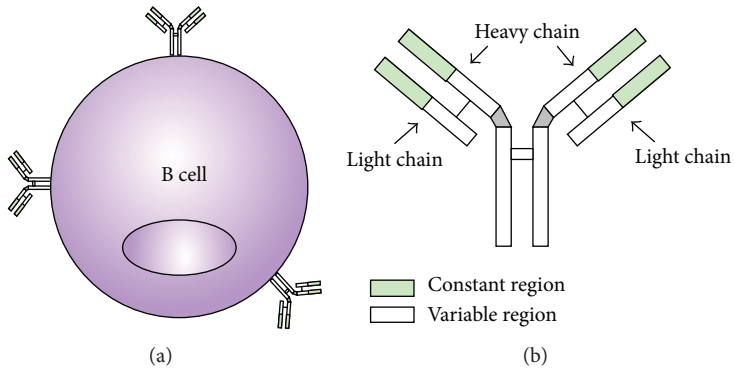


Figure 2: A schematic representation of the antibody.

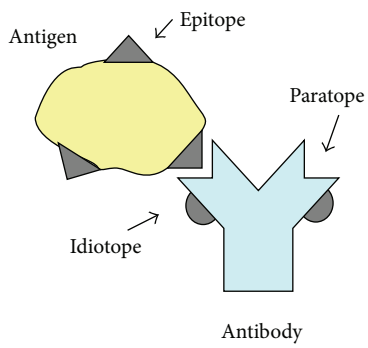


Figure 3: A schematic representation of the antigen-antibody-complex.

An antibody has a Y-shape structure shown in Figure 2. Both arms of the Y-shape consist of two identical heavy and two identical light chains. The tips of the arms are called the variable regions and vary from one antibody to another. The variable regions (paratopes, small segments on antibodies) enable the antibody to match antigen and bind to the epitopes which are small segments on the surface of an antigen. After a binding between an antibody and an antigen's epitope, an antigen-antibody-complex is formed, which results in the deactivation of the antigen [16] shown in Figure 3.

When the antigen-antibody-complex binding is made between B cell and antigen, clonal proliferation (clonal selection) on the B cell then takes place and is strengthened by helper T cells. In clonal proliferation, random changes (mutation) are introduced to the genes, such changes will increase the diversity of the antibody. These higher-affinity matured cells are then selected to enter the pool of memory cells. During clonal proliferation two types of cells are generated: plasma cells and memory cells. The function of memory cells is to proliferate to plasma cells for a faster reaction to frequently encountered antigens and produce antibodies for the antigens. A plasma cell is a B cell that produces antibodies. The fittest clones are those which produce antibodies that bind antigen best (with highest affinity).

3.2. AIS Algorithm

3.2.1. Antibody Encoding Scheme. In AIS algorithm, an antibody (solution) is represented by a string of numbers consisting of the parameters in the optimization problems. Notice that the initial population for each antibody is randomly generated. This encoding method shares the similarities with the evolutionary algorithms (EAs), PSO, and GAs.

3.2.2. Affinity Representation and Measurement. Affinity is the fitness value of antibodies in AIS algorithm. Space-shape concept is developed to measure the complementary space between receptor molecules and antigen, thus providing quantitative information of the interactions (degree of binding) of receptor molecules (antibody) and antigens [15]. Based on this space-shape approach, the generalized shapes of antibody (Ab) and antigens (Ag) in AIS algorithm are defined by $\mathrm{Ab} = \{\mathrm{Ab}_1, \mathrm{Ab}_2, ..., \mathrm{Ab}_L\}$ and $\mathrm{Ag} = \{\mathrm{Ag}_1, \mathrm{Ag}_2, ..., \mathrm{Ag}_L\}$, respectively.

The Ag-Ab affinity is related to their space distance between antibody strings or vectors. The distance D in the Ag-Ab binding vectors can be defined in Euclidean distance (14). An antibody with higher affinity with antigen and lower affinity with other antibodies is better than others:

$$D = \sqrt{\sum_{i=1}^{L} (\mathrm{Ab}_i - \mathrm{Ag}_i)^2}. \tag{14}$$

3.2.3. Clonal Selection. This process in AIS is the selection of a set of antibodies with the highest affinity with a nonself pattern. The selected antibodies are then cloned in an attempt to have a higher binding affinity with the invading antigens. This reproduction selects high-affinity B cells and removes B cells which do not effectively bind with the invading antigen. The number of clones $N_c(i)$ for antibody Ab_i from the antibody population is calculated by $N_c(i) = \mathrm{round}(\lambda n/i)$, where i is the ith highest affinity antibody in the population, n is the number of selected antibodies for cloning, λ is a multiplying factor, and round() is the operator that returns the closest integer.

3.2.4. Somatic Hypermutation. In the proposed AIS, when a B cell is stimulated, a hypermutation mechanism is triggered in the cell's variable region. The mechanism plays a critical role in creating diverse antibody receptors and in increasing antibody affinity and specificity [15]. Each antibody (candidate solution) has an independent mutation rate proportional to its affinity with respect to the nonself antigen, namely that

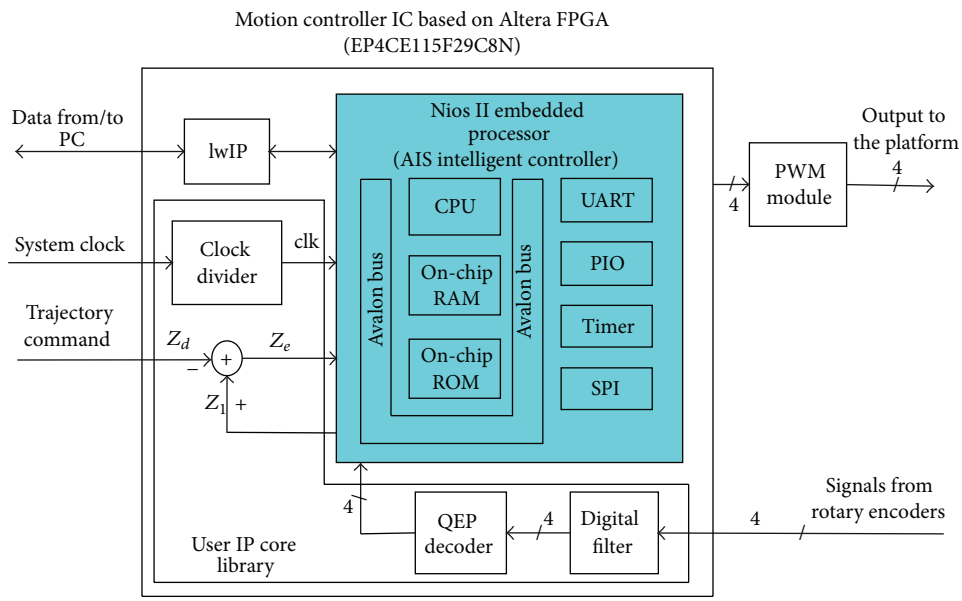


Figure 4: FPGA implementation of the proposed AIS intelligent motion controller.

the hypermutation rate is inversely proportional to the cell affinity.

3.3. Application to Intelligent Motion Control. This subsection aims to employ the AIS computing method in Section 3.2 to design an intelligent motion controller for holonomic omnidirectional mobile robots. In the proposed AIS applied to solve optimal controller problems, the antigen is the problem to solve and the antibody is the generated solution. An antibody represents the control parameters k_{p1}, k_{p2}, k_{p3}, k_{i1}, k_{i2}, k_{i3} in (12), namely, that an antibody is a parameter sequence k_{p1}, k_{p2}, k_{p3}, k_{i1}, k_{i2}, k_{i3}. This optimal feasible sequence will be evolved by the proposed AIS algorithm to achieve trajectory tracking and stabilization for holonomic four-wheeled omnidirectional mobile robots.

Affinity indicates the binding capability between an antibody and an antigen. It determines the quality of the antibody solutions according to the predefined AIS affinity function. Based on the affinity distances in (14) in this optimal control problem, the affinity function (performance index) used to evaluate the antibodies is the following integral square error (ISE) of the four-wheeled holonomic mobile robot given by

$$\mathrm{ISE} = \int_0^t \left(x_e^2(\tau) + y_e^2(\tau) + \theta_e^2(\tau)\right) d\tau. \tag{15}$$

The AIS algorithm for optimal controller design of four-wheeled holonomic mobile robots is described by the following steps.

Step 1. Initialize the AIS algorithm and number of iterations.

Step 2. Randomly generate an antibody population of size $(\mathrm{Ab}_1, \ldots, \mathrm{Ab}_s)$. Each antibody contains genes to represent the control parameters $k_{p1}, k_{p2}, k_{p3}, k_{i1}, k_{i2}, k_{i3}$.

Step 3. Calculate the affinity value for all the antibodies using the affinity function (15).

Step 4. Generate clone set for the best n antibodies. The number of copies is proportional to their affinities: the higher the affinity, the larger the clone size.

Step 5. Update antibodies.

Step 6. Perform hypermutation process to increase the diversity and generate the matured clone set.

Step 7. Check the stopping criterion. If the stop criterion is not matched, go to Step 3 and set $t = t + 1$; otherwise, output the optimal solution (best memory cell) and its corresponding control parameters and stop the algorithm.

Remark 1. This proposed AIS algorithm can be easily extended to not only design the optimal dynamic controllers with parameter variations and uncertainties [14] but also develop the intelligent polar-space motion controllers for holonomic robots [3, 4].

4. FPGA Implementation of the AIS Intelligent Motion Controller

This subsection is devoted to designing a low-cost embedded AIS motion controller of the holonomic omnidirectional mobile robot in order to achieve the control law (12). Figure 4 depicts the architecture of the Altera FPGA implementation for the proposed AIS-based mobile robot motion controller using SoPC technology and hardware/software codesign approach. Both motion control law in (12) and AIS tuning

algorithm have been implemented into the 32-bit Nios II processor whose numerical precision and computation speed are high enough to realize the control law and AIS algorithm. The hardware user IP cores (custom logic), including quadrature-encoder-pulse (QEP), pulse width modulation (PWM), and digital filter for this robotics application, have been developed by VHDL (VHSIC Hardware Description Language). The software AIS-based motion controller and hardware custom logic are connected to the system interconnect fabric via Avalon bus for developing the pragmatic AIS-based intelligent motion controller in one FPGA chip.

With the obtained QEP signals from the encoders, the current position and orientation $Z_1(t) = [x(t) \;\; y(t) \;\; \theta(t)]^T$ of the holonomic mobile robot can be dead-reckoned by the embedded soft-core CPU Nios II. The optimized AIS controller's output $\omega = [\omega_1(t) \;\; \omega_2(t) \;\; \omega_3(t) \;\; \omega_4(t)]^T$ is generated to the PWM module and then steers the four DC brushless motors mounted on the four omnidirectional wheels. Worthy of mention is that a real-time operating system (RTOS) MicroC/OS-II was ported into the same FPGA chip to construct the Ethernet network and handle the data communication with PC via TCP/IP protocol. Moreover, the embedded soft-core Nios II processor works with the lwIP (lightweight IP) for the Ethernet connectivity, thereby significantly reducing the FPGA resource usage. Compared with the PC-based AISs [15–20], the proposed FPGA-based AIS intelligent controller is more efficient and practical for robotics applications. The used FPGA chip is Altera Cyclone IV EP4CE115F29C8N with 114,480 LEs, 525 user I/O pins, 3,888 embedded memory (Kbits), 4 PLLs (Phase Lock Loop), and an embedded soft-core Nios II 32-bit RISC (Reduced Instruction Set Computer) processor.

4.1. FPGA Implementation of the AIS-Based Motion Controller. The embedded AIS-based controller of the holonomic mobile robot is presented in Figure 4, in which the embedded processor in FPGA is employed to perform the AIS tuning algorithm and motion control law. Figure 5 depicts the flowchart of the embedded AIS-based controller program, where the main program and the interrupt service routine (ISR) for the intelligent motion controller were coded in C/C++ language. In the main program, the AIS algorithm is executed by the processor to offline-tune the control parameters, k_{p1}, k_{p2}, k_{p3}, k_{i1}, k_{i2}, k_{i3}. This optimized parameter sequence will be employed in the ISR motion controller program to obtain optimal control performance. Compared with the nonoptimal controller [1–4], this autotuning approach takes the advantages of optimal performance and high efficiency.

With the optimal control parameters obtained from the AIS tuner, the embedded processor calculates the current position and orientation of the holonomic mobile robot from the QEP circuit via dead-reckoning approach and then obtains the values of Z_1 and Z_e. Finally, the output of the intelligent AIS control law is sent out to the PWM module, thus steering the four holonomic omnidirectional wheels. Through the Altera Nios II integrated development environment (IDE), the AIS-based controller program is cross-compiled and downloaded into the SDRAM via JTAG interface.

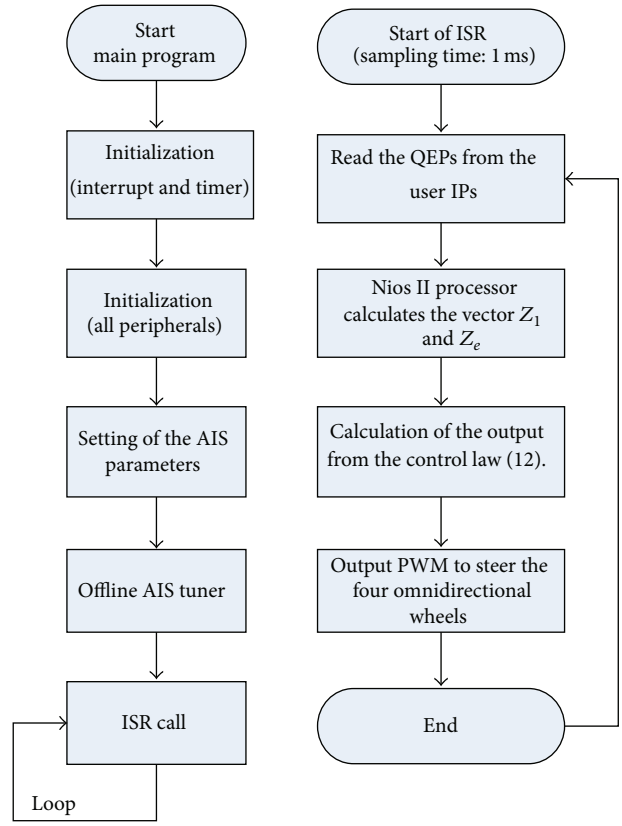


Figure 5: Flow chart of the proposed AIS parameter tuner.

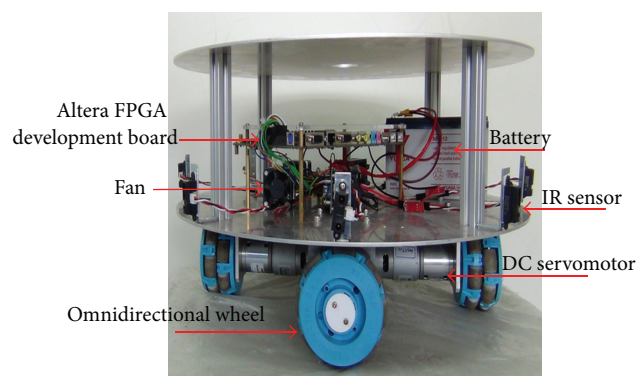


Figure 6: Picture of the experimental holonomic omnidirectional mobile robot.

4.2. FPGA Implementation of the AIS Parameter Tuner. As mentioned in Section 4.1, the AIS parameter autotuner plays a key role in the proposed optimal motion controller. This subsection presents the FPGA implementation of the AIS parameter tuner, aiming at providing the optimized control sequence k_{p1}, k_{p2}, k_{p3}, k_{i1}, k_{i2}, k_{i3}. The AIS auto-tuner is implemented in embedded processor to evolve the optimal control sequence via the clonal selection and hypermutation operations described in Section 3. This processor works

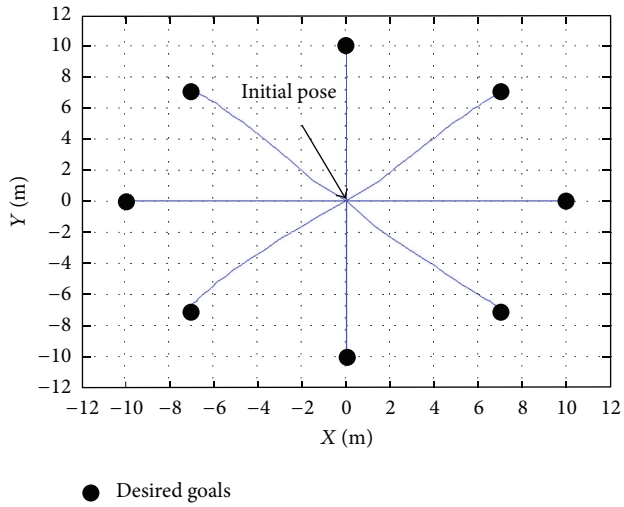


Figure 7: Experimental results of the proposed AIS-based motion controller for achieving stabilization.

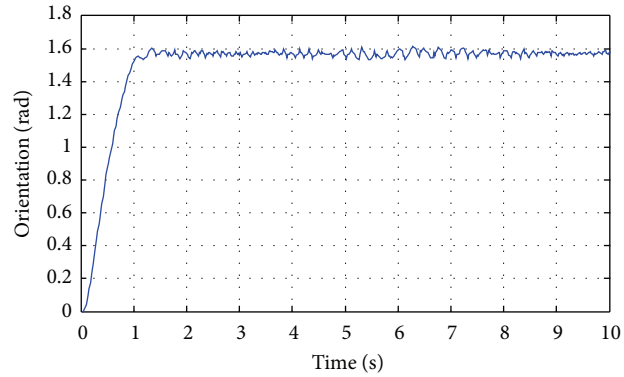


Figure 8: Illustration of the orientation behavior moving towards the desired orientation of $\pi/2$ in the case $n = 0$.

with the hardware custom logic IPs in one FPGA chip to achieve the stabilization and trajectory tracking tasks by using hardware/software technique and SoPC technology. This efficient AIS tuner searches for the optimal solutions, resulting in an optimal sequence to the motion controller for the four-wheeled omnidirectional mobile robot. This FPGA approach is very useful to construct an experimental AIS-based holonomic mobile robot to realize the proposed methods.

Remark 2. This proposed FPGA-based AIS algorithm can be easily extended to design optimal motion controllers in N-wheeled omnidirectional mobile robots.

5. Experimental Results and Discussion

The aims of the experiments are to examine the effectiveness and performance of the proposed SoPC-based AIS intelligent motion control law (12) to the four-wheeled holonomic omnidirectional mobile robot. In order to illustrate the benefits of the proposed AIS-based optimal motion controller, these results are compared with the tracking performance in conventional non-optimal controllers [1–4], GA-based controllers [21–23], and PSO-based controllers [24–26]. The optimal sequence is obtained from the AIS intelligence computing with the following parameters: $\lambda = 2$, $\rho = 10$, $n = 6$.

5.1. System Architecture of the Experimental Holonomic Mobile Robot. The aim of the following experiments is to examine the effectiveness and performance of the proposed AIS-based embedded optimal controller by constructing an experimental holonomic mobile robot shown in Figure 6. The holonomic omnidirectional wheels are driven by four DC brushless servomotors with four mounted encoders of 300 pulses per revolution to provide QEP information. The proposed AIS tuning method and control law are realized using C/C++ code in the Altera Nios II embedded processor incorporating with the developed robotics custom logic IPs. The FPGA chip integrated the embedded soft-core processor, RTOS, lwIP, and VHDL-based IP circuits for performing the SoPC-based AIS control law of the holonomic mobile robot. All the experiments were conducted with the system parameters: $L = 19.4$ cm and $r = 5.08$ cm.

5.2. Point Stabilization. The first experiment was conducted to investigate the regulation performance of the proposed AIS intelligent control law (12). The initial pose of the omnidirectional mobile robot was assumed at the origin; that is, $[x_0 \;\; y_0 \;\; \theta_0] = [\,0\,\text{m} \;\; 0\,\text{m} \;\; 0\,\text{rad}]$, and the desired final eight goal postures are located on the unit circle, given by $[10\cos(n\pi/4)\,\text{m} \;\; 10\sin(n\pi/4)\,\text{m} \;\; (\pi/2)\,\text{rad}]^T$, $n = 0, 1, 2, \ldots, 7$.

Figure 7 depicts all the experimental trajectories of the omnidirectional mobile robot from the origin to the eight goal poses, and Figure 8 shows the heading behavior of the proposed stabilization law for the mobile robot moving towards the desired orientation $\pi/2$ in the case $n = 0$. Through experimental results, the holonomic mobile robot with the proposed AIS stabilization method has been shown capable of reaching the desired postures. Comparing this stabilization result with the one in non-optimal controllers [1–4], this AIS-tuning approach optimizes the controller's performance and ensures the stability without manually tuning the control parameters.

In order to exhibit the merit of the proposed AIS tuning approach over conventional GA and PSO, Figure 9 presents the evolutions of performance index (fitness value) for the proposed SoPC-based AIS-tuning controller and the two conventional controllers to achieve point stabilization. As shown in Figure 9, the proposed FPGA-based AIS computational intelligence converges to the optimal solution with better performance index. Through these experimental results, the holonomic mobile robot with the FPGA-based AIS optimal controller is superior to the conventional controllers.

5.3. Elliptic Trajectory Tracking. The elliptic trajectory tracking experiment is aimed to explore how the proposed

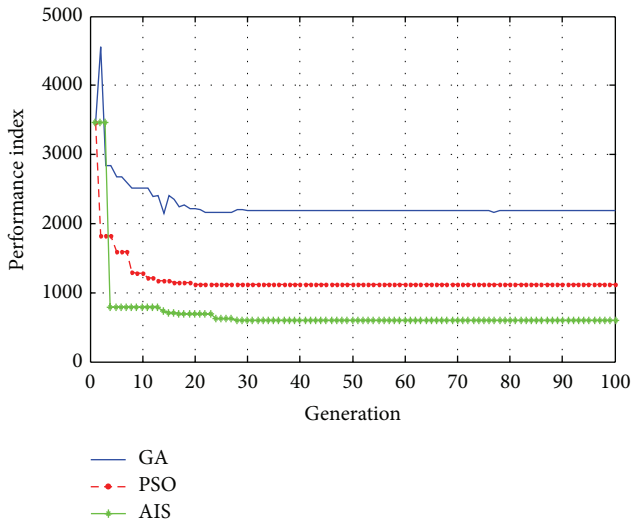


Figure 9: The evolutions of performance index in the proposed AIS, GA and PSO controllers to achieve point stabilization.

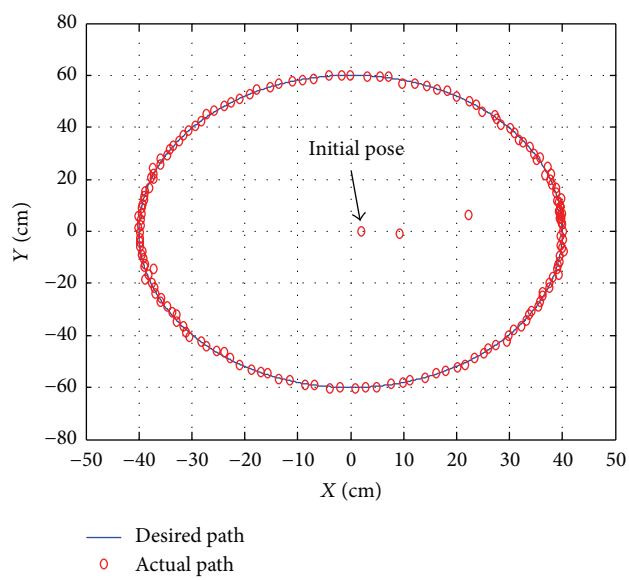


Figure 10: Experimental result of the elliptic trajectory tracking.

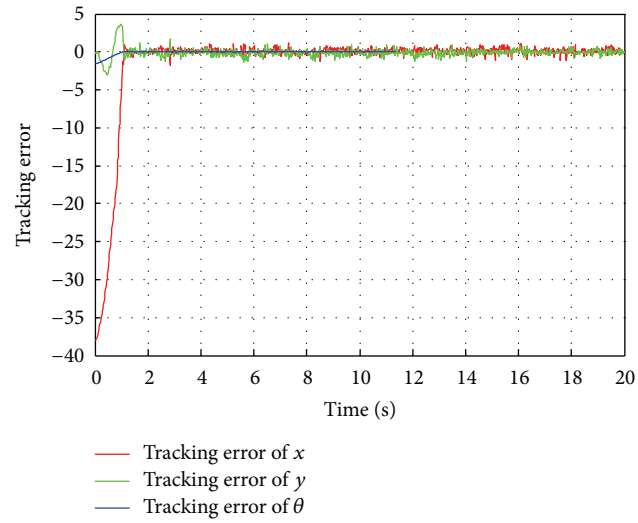


Figure 11: Tracking errors for the elliptic trajectory tracking.

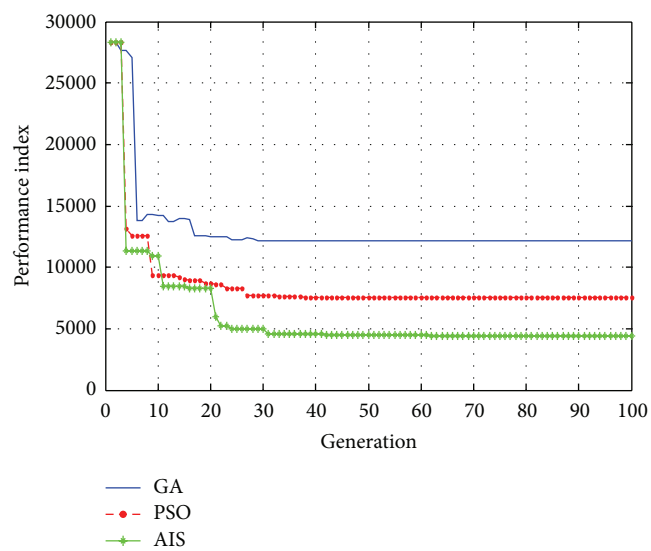


Figure 12: The evolutions of performance index of the proposed AIS, GA, and PSO controllers to achieve elliptic trajectory tracking.

controller (12) steers the holonomic mobile robot to exactly track an elliptic trajectory described by $[x_r \;\; y_r \;\; \theta_r] = [40\cos w_r t\,(\text{cm}) \;\; 60\sin w_r t\,(\text{cm}) \;\; 0\,(\text{rad})]$, $w_r = 0.2\,\text{rad/sec}$. The experiment assumed that the mobile robot got started at $[x_0 \;\; y_0 \;\; \theta_0] = [1\,\text{m} \;\; 0\,\text{m} \;\; 0\,\text{rad}]$. Figure 10 presents the experimental result for elliptic trajectory tracking of the mobile robot. The tracking errors for the elliptic trajectory are depicted in Figure 11.

Figure 12 presents the evolutions of performance index in the proposed AIS, GA, and PSO controllers to achieve trajectory tracking. These results indicate that the proposed FPGA-based AIS intelligent controller (12) is capable of successfully steering the holonomic omnidirectional mobile robot to track the elliptic trajectory. The proposed FPGA-based AIS computational intelligence outperforms the conventional GA and PSO approaches in designing optimal motion controller for the four-wheeled holonomic mobile robots.

6. Conclusion

This paper has presented an intelligent motion controller using FPGA-based AIS computational intelligence to achieve trajectory tracking and stabilization of a holonomic omnidirectional mobile robot with four independent driving wheels equally spaced at 90 degrees from one another. Based on the kinematic model, an intelligent kinematic controller has been synthesized via AIS computing paradigm. Both AIS parameter tuner and kinematic motion controller are integrated in one FPGA chip to efficiently construct an experimental

holonomic mobile robot. Through experimental results, the proposed AIS-based optimal motion controller has been shown to achieve stabilization and trajectory tracking successfully. These results indicate that the proposed FPGA-based intelligent AIS optimal controller outperforms the non-optimal controllers and the conventional optimal controllers.

Acknowledgment

The authors gratefully acknowledge financial support from the National Science Council, Taiwan, Taiwan, under grant under grant NSC 102-2221-E-197-023.

References

[1] T. Kalmár-Nagy, R. D'Andrea, and P. Ganguly, "Near-optimal dynamic trajectory generation and control of an omnidirectional vehicle," *Robotics and Autonomous Systems*, vol. 46, no. 1, pp. 47–64, 2004.

[2] H.-C. Huang, C.-C. Tsai, and S.-C. Lin, "Adaptive polar-space motion control for embedded omnidirectional mobile robots with parameter variations and uncertainties," *Journal of Intelligent and Robotic Systems*, vol. 62, no. 1, pp. 81–102, 2011.

[3] D. Chwa, "Sliding-mode tracking control of nonholonomic wheeled mobile robots in polar coordinates," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 4, pp. 637–644, 2004.

[4] H.-C. Huang and C.-C. Tsai, "Adaptive robust control of an omnidirectional mobile platform for autonomous service robots in polar coordinates," *Journal of Intelligent and Robotic Systems*, vol. 51, no. 4, pp. 439–460, 2008.

[5] G. Indiveri, "Swedish wheeled omnidirectional mobile robots: kinematics analysis and control," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 164–171, 2009.

[6] Z.-P. Jiang and H. Nijmeijer, "A recursive technique for tracking control of nonholonomic systems in chained form," *IEEE Transactions on Automatic Control*, vol. 44, no. 2, pp. 265–279, 1999.

[7] T.-C. Lee, K.-T. Song, C.-H. Lee, and C.-C. Teng, "Tracking control of unicycle-modeled mobile robots using a saturation feedback controller," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 2, pp. 305–318, 2001.

[8] T.-H. S. Li, S.-J. Chang, and Y.-X. Chen, "Implementation of human-like driving skills by autonomous fuzzy behavior control on an FPGA-based car-like mobile robot," *IEEE Transactions on Industrial Electronics*, vol. 50, no. 5, pp. 867–880, 2003.

[9] E. Hashemi, M. G. Jadidi, and O. B. Babarsad, "Trajectory planning optimization with dynamic modeling of four wheeled omni-directional mobile robots," in *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '09)*, pp. 272–277, December 2009.

[10] K.-S. Byun, S.-J. Kim, and J.-B. Song, "Design of a four-wheeled omnidirectional mobile robot with variable wheel arrangement mechanism," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 720–725, Washington, DC, USA, May 2002.

[11] O. Purwin and R. D'Andrea, "Trajectory generation for four wheeled omnidirectional vehicles," in *Proceedings of the American Control Conference (ACC '05)*, pp. 4979–4984, Portland, Ore, USA, June 2005.

[12] C.-C. Shing, P.-L. Hsu, and S.-S. Yen, "T-S fuzzy path controller design for the omnidirectional mobile robot," in *Proceedings of the 32nd Annual Conference on IEEE Industrial Electronics (IECON '06)*, pp. 4142–4147, November 2006.

[13] T.-H. S. Li, C.-Y. Chen, H.-L. Hung, and Y.-C. Yeh, "A fully fuzzy trajectory tracking control design for surveillance and security robots," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '08)*, pp. 1995–2000, Singapore, October 2008.

[14] C.-C. Tsai, Z.-R. Wu, Z.-C. Wang, and M.-F. Hisu, "Adaptive dynamic motion controller design for a four-wheeled omnidirectional mobile robot," in *Proceeding of the International Conference on System Science and Engineering (ICSSE '10)*, pp. 233–238, twn, July 2010.

[15] L. N. de Castro and J. I. Timmis, "Artificial immune systems as a novel soft computing paradigm," *Soft Computing*, vol. 7, no. 8, pp. 526–544, 2003.

[16] Z. Zhang and T. Xin, "Immune algorithm with adaptive sampling in noisy environments and its application to stochastic optimization oroblems," *IEEE Computational Intelligence Magazine*, vol. 2, no. 4, pp. 29–40, 2007.

[17] T.-C. Chen and P.-S. You, "Immune algorithms-based approach for redundant reliability problems with multiple component choices," *Computers in Industry*, vol. 56, no. 2, pp. 195–205, 2005.

[18] J.-T. Tsai and J.-H. Chou, "Design of optimal digital IIR filters by using an improved immune algorithm," *IEEE Transactions on Signal Processing*, vol. 54, no. 12, pp. 4582–4595, 2006.

[19] S.-W. Lin and S.-C. Chen, "Parameter tuning, feature selection and weight assignment of features for case-based reasoning by artificial immune system," *Applied Soft Computing Journal*, vol. 11, no. 8, pp. 5042–5052, 2011.

[20] D. H. Kim, "Tuning of 2-DOF PID controller by immune algorithm," in *Proceedings ofthe 2002 Congress on Evolutionary Computation*, vol. 1, pp. 675–680, 2002.

[21] J. C. Gallagher, S. Vigraham, and G. Kramer, "A family of compact genetic algorithms for intrinsic evolvable hardware," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 2, pp. 111–126, 2004.

[22] C.-K. Ting, C.-N. Lee, H.-C. Chang, and J.-S. Wu, "Wireless heterogeneous transmitter placement using multiobjective variable-length genetic algorithm," *IEEE transactions on systems, man, and cybernetics B*, vol. 39, no. 4, pp. 945–958, 2009.

[23] S. Yue, D. Henrich, W. L. Xu, and S. K. Tso, "Point-to-point trajectory planning of flexible redundant robot manipulators using genetic algorithms," *Robotica*, vol. 20, no. 3, pp. 269–280, 2002.

[24] Z.-L. Gaing, "A particle swarm optimization approach for optimum design of PID controller in AVR system," *IEEE Transactions on Energy Conversion*, vol. 19, no. 2, pp. 384–391, 2004.

[25] R.-J. Wai, J.-D. Lee, and K.-L. Chuang, "Real-time PID control strategy for maglev transportation system via particle swarm optimization," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 2, pp. 629–646, 2011.

[26] Y. Song, Z. Chen, and Z. Yuan, "New chaotic PSO-based neural network predictive control for nonlinear process," *IEEE Transactions on Neural Networks*, vol. 18, no. 2, pp. 595–600, 2007.

[27] S. Sánchez-Solano, A. J. Cabrera, I. Baturone, F. J. Moreno-Velo, and M. Brox, "FPGA implementation of embedded fuzzy controllers for robotic applications," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 4, pp. 1937–1945, 2007.

[28] S. T. Pan and X. Y. Li, "An FPGA-based embedded robust speech recognition system designed by combining empirical mode

decomposition and a genetic algorithm," *IEEE Transactions OnInstrumentation and Measurement*, vol. 61, no. 9, pp. 2560–2572, 2012.

[29] D. Zhang and H. Li, "A stochastic-based FPGA controller for an induction motor drive with integrated neural network algorithms," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 2, pp. 551–561, 2008.

[30] N. G. Johnson-Williams, R. S. Miyaoka, X. Li, T. K. Lewellen, and S. Hauck, "Design of a real time FPGA-based three dimensional positioning algorithm," *IEEE Transactions on Nuclear Science*, vol. 58, no. 1, pp. 26–33, 2011.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 581754, 12 pages
http://dx.doi.org/10.1155/2013/581754

Research Article

Fatigue Life Prediction Using Simplified Endurance Function Model

M. Kamal, M. M. Rahman, and M. S. M. Sani

Faculty of Mechanical Engineering, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia

Correspondence should be addressed to M. M. Rahman; mustafizur@ump.edu.my

Received 3 August 2013; Revised 14 October 2013; Accepted 23 October 2013

Academic Editor: Sheng-yong Chen

A methodology is proposed to apply an endurance function model with a genetic algorithm to estimate the fatigue life of notched or smooth components. The endurance function model is based on stress tensor invariants and deviatoric stress invariants. In the proposed methodology, FEA is used to simplify the application of the endurance function model. Experimental results from published literature are considered for the case studies to evaluate the proposed methodology. The results show that the proposed methodology simplified the application of the endurance function model, particularly by reducing the need for notch sensitivity factors, and the stress invariants can be calculated directly from the stresses at the critical point. The comparison with experimental results shows that, with proper calibration, the model can predict fatigue life accurately.

1. Introduction

Since the investigations by Wohler in 1860, fatigue experiments and predictions have played a major role in mechanical design [1, 2], and researchers investigating the fatigue problem have made huge efforts in order to devise sound methodologies suitable for safely assessing mechanical components subjected to time-variable loadings [3–7]. It is acknowledged globally that correctly estimating fatigue damage in real components is a complex process involving a high number of different variables that have to be properly taken into account in order to avoid unwanted and dangerous failures [8]. Any reliable fatigue assessment technique should be able to efficiently and simultaneously model the damaging effect of nonzero superimposed static stresses, the degree of multiaxiality of the stress field and the role of stress concentration phenomena [9]. Especially in the case of cyclic or random multiaxial loading histories, the fatigue assessment is difficult to correctly perform since damage accumulation depends on all the components of the stress tensor and their variation during the whole phenomenon [9, 10]. To ensure that their results are close to reality, the calibration of such an engineering fatigue assessment method should be based on pieces of experimental information that can be easily obtained through tests run in accordance with the pertinent standard codes [8, 9, 11–13]. The stress analysis is conducted to correctly estimate fatigue damage by directly postprocessing simple linear elastic finite element (FE) models [14–17].

To deal with the fatigue life assessment problem of structural components under multiaxial load histories (proportional or nonproportional, cyclic or random), Brighenti and Carpinteri [8] proposed an endurance function based fatigue life estimation model, based on a continuum damage mechanics formulation [18]. The model does not require any evaluation of a critical plane as it considers the damage accumulated at a point using stress tensor invariants and deviatoric stress invariants, and invariants are not coordinate system specific quantities. Also, as per the continuum mechanics concept, the endurance function is defined as a continuously evolving function with applied loading, so there is no requirement for any conventional loading cycle counting algorithm [8, 18]. The damage (D) is evaluated at a specific point of the structural component through the appropriate endurance function (E) and a suitable expression of damage increment (dD). The fatigue life is assumed to occur when damage D reaches unity. A genetic algorithm (GA) approach is employed to evaluate numerically the several parameters used for characterization of the damage

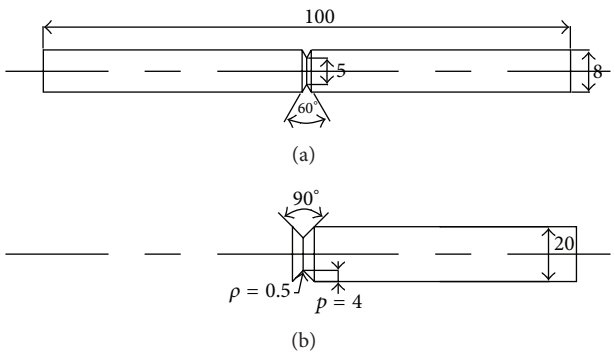


Figure 1: (a) Specimen used for EN3B testing with notch radius 1.25 mm [9]; (b) specimen used for C40 testing [13].

mechanics approach, once the effects of some experimental complex histories are known [8, 19]. In this paper, a methodology has been proposed to predict fatigue life using the endurance function model and GA procedures coupled with finite element analysis. Two steel alloys EN3B (cold rolled low carbon steel) and C40 (carbon steel) are selected for study. Experimental fatigue life data for tension torsion tests from published literature is used [9, 13]. The determination of stress invariants for the endurance function is greatly simplified due to the use of finite element analysis and also helps reduce the number of parameters by one; that is, the stress concentration effect can be avoided as we can obtain the exact value of stresses at the notch. The results show that the above mentioned methodology worked well in the low and medium cycle range, while for high cycles the results are highly conservative.

2. Finite Element Modeling and Analysis

Two sets of experimental data on tension torsion fatigue life on steel alloys EN3B and C40 were considered in this paper [9, 13]. The specimen dimension detail used to obtain the results for each alloy is shown in Figure 1. The three-dimensional model is designed using computer-aided design software and the finite element model is developed utilizing ANSYS software with 10-node tetrahedral elements, to better capture the curved surfaces of the specimen geometry. Dense mesh at the notch root is maintained by the sphere of influence technique, where mesh size control is implemented by defining a spherical volume in which required mesh size is maintained and not in the whole meshed component. Finite element analysis is performed on both specimen geometries with the same respective loading conditions used f; the specimens are of nearly the same geometry or experimental testing (Table 1) and the FEA model is shown in Figure 2. Load is applied as force and moment, which will result in the required applied normal and shear stresses at the net area, as mentioned in Table 1.

2.1. Mesh Sensitivity Analysis. Mesh sensitivity analysis has been performed to obtain the optimum mesh size which will give a good balance between accuracy, processor time, and storage load [19]. Table 2 shows the result of the mesh sensitivity analysis. As the specimens are of similar notch size and shape, the resulting optimum mesh size is also of the same size; mesh sensitivity is not needed for individual specimens. From Table 2, it can be seen that after a mesh size of 0.175 mm the stress values do not change by an appreciable amount, but there is an exponential rise in the number of nodes and elements, which will result in an increase of the processor time and storage requirement without much increase in the accuracy of the stress results. Hence, to get the optimum performance a mesh size of 0.175 mm is selected for meshing both specimen models.

3. Endurance Function Model

Brighenti and Carpinteri [8] proposed an endurance function model on the basis of a continuum damage mechanics approach, with the assumption that the whole fatigue life is crack nucleation dominated and that the fatigue life for crack propagation is negligible with respect to total life. For isotropic materials the endurance function can be expressed as below:

$$\begin{gathered} E(\sigma, S_e) = \left[a_1 \cdot I_1(\sigma) + a_2 \cdot I_2^{1/2}(\sigma) + a_3 \cdot I_3^{1/3}(\sigma) \right. \\ \left. + a_4 \cdot J_2^{1/2}(S_e) + a_5 \cdot J_3^{1/3}(S_e)\right] - \sigma_0 = 0, \\ I_1 = \sigma_1 + \sigma_2 + \sigma_3, \quad I_2 = \sigma_1\sigma_2 + \sigma_2\sigma_3 + \sigma_3\sigma_1, \\ I_3 = \sigma_1\sigma_2\sigma_3, \\ J_2 = \frac{\left(I_1^2 - 3I_2\right)}{3}, \quad J_3 = \frac{\left(2I_1^3 - 9I_1I_2 + 27I_3\right)}{27}, \end{gathered} \tag{1}$$

where a_1 to a_5 and σ_0 are the material constants, I_1, I_2, and I_3 are stress tensor invariants, and J_2, J_3 deviatoric stress invariants are functions of stress tensor σ and effective deviatoric stress tensor S_e, respectively. Here, $S_e = S - S_b$; that is, S is the current applied deviatoric stress tensor and S_b is the back stress tensor which measures the endurance function evolution in the stress space.

Damage is assumed to occur when $E(\sigma, S_e) > 0$ and no damage occurs when $E(\sigma, S_e) \leq 0$, and an increment in damage will happen when dE (increment in endurance function) > 0. To define dE properly, it is specified that if there is a case where the stress value at point i results in $E(\sigma_i, S_{e,i}) > 0$ and the previous point results in $E(\sigma_{i-1}, S_{e,i-1}) < 0$, the quantity $E(\sigma_{i-1}, S_{e,i-1})$ is set equal to zero, which in turn results in keeping dE always greater than zero.

The damage D is evaluated by considering the progressive accumulation of damage increments; that is, at each load step the damage increment is equal to or greater than zero $dD \geq 0$ and consequently the material damage D is a non-decreasing positive function; that is, $D \geq 0$ [3]. And final collapse occurs when D reaches unity ($D = 1$). The damage

Table 1: Experimental loading conditions and fatigue life of EN3B specimen having notch radius 1.25 mm and C40 specimen having notch radius 0.5 mm.

Normal stress at net area σ_a (MPa)	Shear stress at net area τ_a (MPa)	Load ratio (R)	Phase difference (°)	Cycles to failure (N_f) $\times 10^6$
		EN3B [9]		
275	158.8	−1	0	0.046
259.6	155.9	−1	0	0.083
230	132.8	−1	0	0.2
200	115.5	−1	0	0.44
190	109.7	−1	0	1.4
180	103.9	−1	0	2.17
285	164.5	−1	90	0.032
270	155.9	−1	90	0.059
260	150.1	−1	90	0.31
250	144.3	−1	90	0.079
230	132.8	−1	90	0.24
200	115.5	−1	90	2.1
		C40 [13]		
200	200	−1	0	0.027
179	179	−1	0	0.22
160	160	−1	0	0.072
129.75	129.75	−1	0	0.18
118.8	118.8	−1	0	0.44
101	101	−1	0	2.0
199.7	199.7	−1	90	0.011
180	180	−1	90	0.014
160.25	160.25	−1	90	0.041
140	140	−1	90	0.28
129.65	129.65	−1	90	0.11
119.5	119.5	−1	90	0.94
109.3	109.3	−1	90	1.11
99.6	99.6	−1	90	2.0
158	158	0	0	0.026
138.5	138.5	0	0	0.047
119	119	0	0	0.15
99.55	99.55	0	0	0.35
89.4	89.4	0	0	0.43
79.72	79.72	0	0	0.79
67.9	67.9	0	0	2.0
158	158	0	90	0.021
138.75	138.75	0	90	0.036
119.3	119.3	0	90	0.085
99.25	99.25	0	90	0.087
89.55	89.55	0	90	0.34
66.8	66.8	0	90	2.0

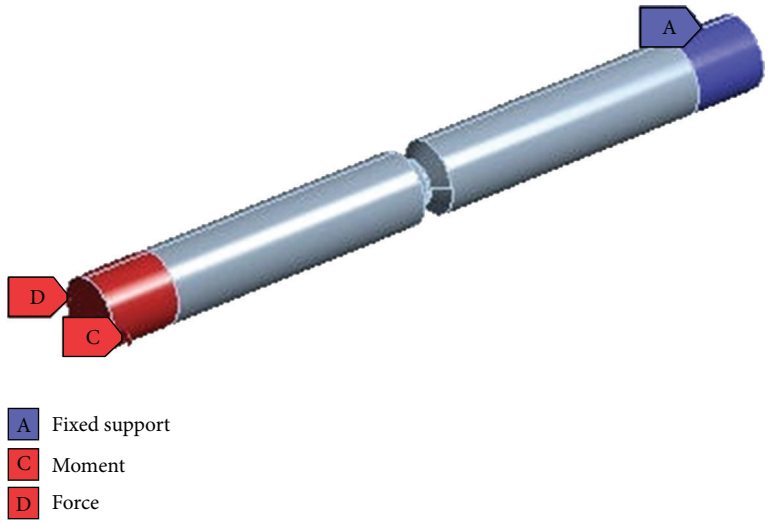


Figure 2: Structural analysis model of specimen.

Table 2: Mesh sensitivity analysis results.

Mesh size (mm)	von mises stress (MPa)	Tresca stress (MPa)	Max principal (MPa)	Number of nodes	Number of elements
0.1	562.96	575.72	576.27	587762	409838
0.125	562	573.85	574.53	414193	287994
0.15	561.1	572.8	574	260754	181331
0.175	560.78	572.65	575.42	176749	122197
0.2	554.79	565.7	569.95	144013	99252

rate dD is assumed to depend on the current value of E as well as dE, with the relationship between dD and dE as follows:

$$dD = A \cdot E^B \cdot dE, \tag{2}$$

where A and B are material constants. The stress gradient effect is taken into account by inserting a reducing factor G into (2):

$$dD = \left[A \cdot E^B \cdot dE\right] \cdot G. \tag{3}$$

Here,

$$G = e^{-V \cdot \gamma}, \tag{4}$$

where G is the notch gradient correction factor which depends on V (material constant) and the stress field parameter γ, which represents the stress gradient absolute value at the notch root. The evolution of deviatoric back stress S_b is assumed to follow the relationship below:

$$dS_b = \begin{cases} C \cdot dE^h \cdot (S - S_b) & \text{if } dD > 0 \\ 0 & \text{if } dD = 0, \end{cases} \tag{5}$$

where C and h are material parameters.

3.1. Application of Genetic Algorithm for Parameter Estimation. A genetic algorithm (GA) is used to find the optimum values of these parameters. Such algorithms (random stochastic methods of global optimization) are used to minimize or maximize an objective function chosen for a given problem to be solved. Such an approach can be useful to evaluate the model parameters (in the context of model parameter tuning), once the response of the physical system to a given input is known [20]. GAs have some advantages with respect to classical techniques, as they allow us to handle problems with multiple minima and nonconvexity properties, thus avoiding numerical instability and missing the global optimum [21]. A GA can handle any kind of objective function and creates a population of solutions and applies genetic operators, such as selection, mutation, crossover, and elitism to evolve the solutions in order to find the best ones [22], by iteratively repeating the "evolution procedure" until a given tolerance is attained [23]. In order to apply the endurance function model, values of 11 parameters that need to be evaluated, as defined in (1)–(5). If the fatigue life N_f is known for the generic multiaxial stress history where damage D reaches unity, a prediction error can be defined as follows:

$$e = D\left(a_1, a_2, \ldots, N_f\right) - 1, \tag{6}$$

where $D(a_1, a_2, \ldots, N_f)$ is the damage evaluated at N_f. The values of the model parameters can be found by minimizing this error function using the GA procedure [8].

4. Simplified Endurance Function Model

The two materials, EN3B and C40 steels, are used in this study with six sets of fatigue life data (Table 1), under cyclic multiaxial in-phase and out of phase loadings. Specimens are notched specimens as shown in Figure 1. The mechanical characteristics of both steels are as follows in Table 3 [8, 9].

Table 3: Mechanical properties of EN3B and C40 steels.

Material name	Young's modulus (GPa)	Yield stress (MPa)	Ultimate tensile strength (MPa)	Loading condition		Fatigue limit (MPa) (at cycles)
			R-Phase			
En3B [9]	208.5	653	676	−1	0	192.4 (10^6 cycles)
				−1	90	188.2 (10^6 cycles)
C40 [13]	206	537	715	−1	0	101 (2×10^6 cycles)
				−1	90	99.6 (2×10^6 cycles)
				0	0	67.9 (2×10^6 cycles)
				0	90	66.8 (2×10^6 cycles)

Table 4: Parameters at of calibration points for endurance function model obtained from GA for EN3B and C40.

Load (MPa) σ	τ		a_1	a_2	a_3	a_4	a_5	σ_0	A	B
				EN3B ($R = -1$ and phase = 0°)						
275	158.8	0.5107	1.004	0.2313	1.129	1.197	180	5.348×10^{-9}	0.2299	
180	103.9	0.4001	0.5233	0.4113	0.7745	0.8194	166.5	1.009×10^{-9}	0.08192	
				EN3B ($R = -1$ and phase = 90°)						
285	164.5	0.4255	1.19	0.3625	1.076	0.7991	141.4	7.056×10^{-9}	0.2389	
200	115.5	0.2708	0.327	0.2189	0.7392	0.4868	149.7	2.882×10^{-9}	0.07962	
				C40 ($R = -1$ and phase = 0°)						
200	200	0.1412	0.3676	0.4504	1.404	−0.3757	80.18	8.034×10^{-9}	0.4285	
101	101	0.4972	0.7769	0.3194	1.44	0.9721	85.33	1.485×10^{-10}	0.2322	
				C40 ($R = -1$ and phase = 90°)						
199.7	199.7	0.4358	0.5965	0.6721	1.428	1.165	76.7	2.868×10^{-8}	0.2134	
99.6	99.6	0.2615	0.2907	0.4818	1.345	−0.2697	79.02	1.032×10^{-9}	0.1377	
				C40 ($R = 0$ and phase = 0°)						
158.1	158.1	0.7446	0.2931	0.432	1.731	0.2959	58.4	8.05×10^{-9}	0.224	
67.9	67.9	0.05374	0.5927	0.6035	0.9929	0.3145	61.08	1.609×10^{-9}	0.03816	
				C40 ($R = 0$ and phase = 90°)						
158	158	0.4721	0.4814	0.5829	1.655	0.7878	54.44	7.373×10^{-9}	0.2677	
66.8	66.8	0.1452	0.6464	0.572	0.9294	0.6613	61.83	1.198×10^{-9}	0.1313	

Initially linear stress analysis of specimens was conducted at every set of applied stress mentioned in Table 1, and all three principal stresses ($\sigma_1, \sigma_2, \sigma_3$) were recorded at the notch root where the highest value of maximum principal stress is occurring [24, 25]. Neuber elastic plastic correction is applied where stresses are found to be above yield strength [2, 14, 26]. Stress invariants (I_1, I_2, I_3, J_2, and J_3) are then calculated from the following relationships depending on principal stresses after Neuber correction is applied where necessary [27].

The endurance function parameters, where G is the notch gradient correction factor, becomes $G = 1$ due to the fact that FEA gives the value of stress at the notch root directly. This in return nullifies the requirement to determine parameter V and the stress field parameter γ, thus reducing the number of endurance function parameters by one. For both of the considered materials EN3B and C40 steel, the cyclic behavior is assumed to be following a stable hysteresis loop, which in return allows the change in back stress $dS_b = 0$. Thus, there is no need to calculate parameters C and h in (5). So finally, from 11 parameters of the endurance function, the number required is reduced to eight, $a_1, a_2, a_3, a_4, a_5, \sigma_0, A$, and B.

Now, to calculate the value of parameters at calibration points (from all loading conditions considered in this study as per Table 1, the selected ones to calibrate the parameters of endurance function are called calibration points. Table 4 shows the calibration points with calibrated parameter values), GA is used where the error function which is to be minimized is defined as (7), with the assumption that, as the cyclic behavior of the material is stable, the damage in each cycle is the inverse of the fatigue life in the cycles:

$$e = D - \frac{1}{N_f}, \tag{7}$$

where the damage per cycle is calculated from the endurance function. These equations are inserted in the GA algorithm for optimization of the endurance function parameters [28–30]. Figure 3 shows the flow chart of fatigue life estimation process.

The initial values table required for the GA is generated randomly from the ranges of parameters assumed. The range for σ_0 is defined with its upper limit as the conventional fatigue limit [8, 9] and the lower limit is set to be 25% less than the upper limit initially and then modified with ranges of other parameters until the combinations become stable, which results in minimum error. From being stable, it is assumed that the sets of parameter values resulting in minimum error have no significant change in respective parameters. Then, the top three values of parameters predicted for each calibration point are selected with respect to the minimum error which are weighted average on the basis of error [31], and the resulting values of weighted average are characterized as the parameters of the endurance function at the respective calibration point. Table 4 shows the parameters at the calibration points determined using the GA.

5. Results and Discussion

The parameters for all the calibration points calculated using the GA, as shown in Table 4 for both materials and their respective load sets, are used to determine values for other loading conditions in Table 1 by interpolation between the parameter values at calibration points; fatigue life is then estimated using the endurance function model. The predicted fatigue life is reported in Table 5. The first thing to notice from the fatigue life prediction results is that the fatigue life values calculated at the calibration points themselves are not the same as the experimental life (Tables 1, 4, and 5), which should theoretically be the same, as these points are used to calculate the coefficients for the endurance function model. The reason lies in the fact that the parameters calculated at each calibration point are the weighted average of top three values predicted by GA, which in turn allowed the estimation of fatigue life at the calibration point to deviate from the experimental life at that point. But the author suggests that this weighted average should nevertheless be used rather than just relying on one set with minimum error, so as to better capture the trend of the coefficients from the three sets from which the weighted average is being taken. The interpolation of the coefficients of the endurance function for load conditions other than calibration points is introduced, where parameter values are interpolated between the two calibration points parameter values with respect to applied load and calibration point load values, which is a better approximation for estimating the fatigue life than the proposed method of calculating only one set of coefficients [8]. Also, the idea of using the weighted average of the coefficients calculated from different load values in turn creates a bias on the basis of error towards the coefficients with minimum error, which should not be there, as all

Table 5: Predicted fatigue life of EN3B and C40 steel using endurance function model.

Load (MPa)		
Normal stress σ (MPa)	Shear stress τ (MPa)	Predicted cycles to failure $(N_p) \times 10^6$
EN3B		
For $R = -1$ and phase = 0°		
275	158.8	0.042
259.6	155.9	0.068
230	132.8	0.18
200	115.5	0.59
190	109.7	0.99
180	103.9	1.83
For $R = -1$ and phase = 90°		
285	164.5	0.032
270	155.9	0.054
260	150.1	0.076
250	144.3	0.11
230	132.8	0.24
200	115.5	0.97
C40		
For $R = -1$ and phase = 0°		
200	200	0.027
179	179	0.032
160	160	0.045
129.75	129.75	0.11
118.8	118.8	0.18
101	101	1.75
For $R = -1$ and phase = 90°		
199.7	199.7	0.0092
180	180	0.014
160.25	160.25	0.023
140	140	0.051
129.65	129.65	0.069
119.5	119.5	0.14
109.3	109.3	0.31
99.6	99.6	1.2
For $R = 0$ and phase = 0°		
158	158	0.025
138.5	138.5	0.048
119	119	0.1
99.55	99.55	0.23
89.4	89.4	0.38
79.72	79.72	0.65
67.9	67.9	0.14
For $R = 0$ and phase = 90°		
158	158	0.02
138.75	138.75	0.035
119.3	119.3	0.066
99.25	99.25	0.14
89.55	89.55	0.21
66.8	66.8	0.85

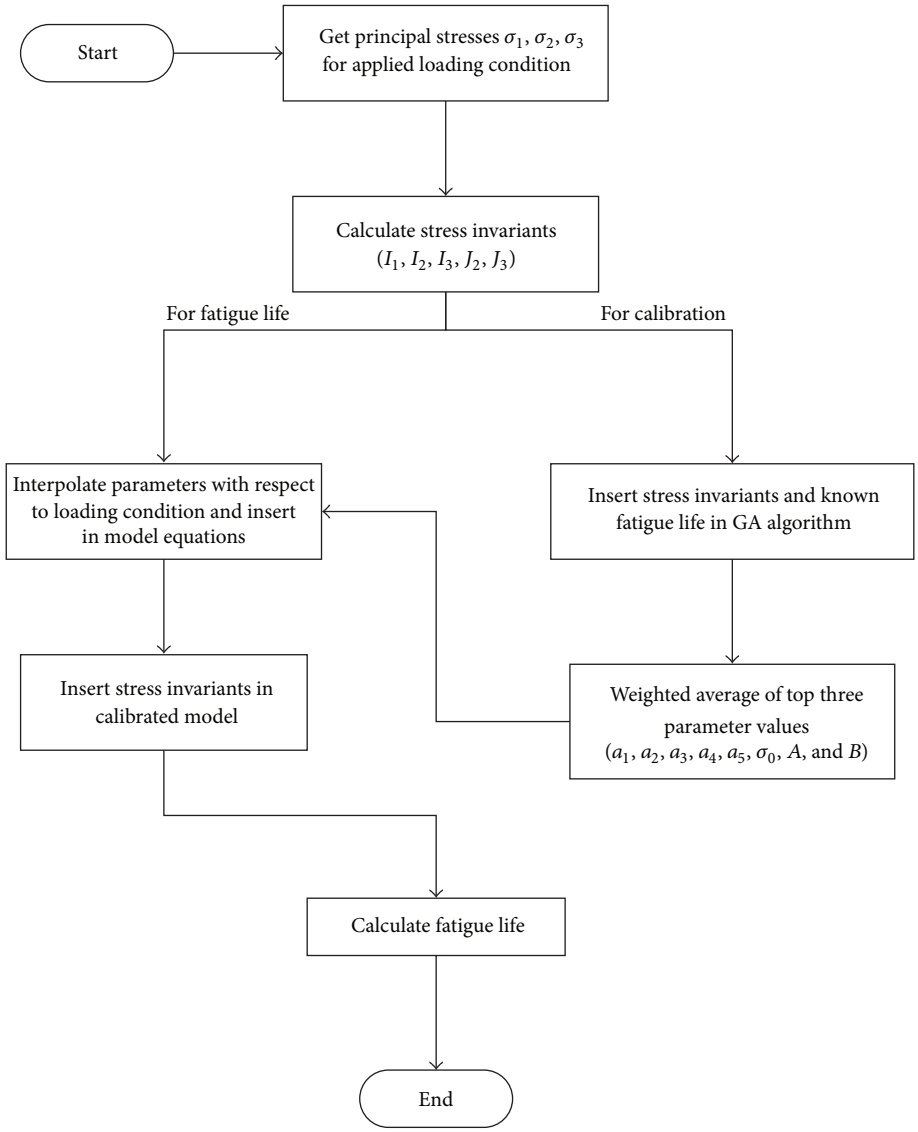


Figure 3: Flow chart of fatigue estimation process.

the calibration load points are experimental values and have equal weight.

This also proves the applicability of interpolation between the coefficients, as this keeps the weight of the coefficients at calibration load points the same, with estimates at other points following the trend of changing the coefficients with the load values as shown in Figure 4. The stress life curves from the predicted and experimental life data for EN3B are shown in Figure 6 and for C40 steel in Figure 7. From the results, it can be seen that the endurance function with the proposed application methodology shows good agreement with experimental data for both materials in the case of in-phase loading. Brighenti et al. [32] also reported that for different materials the endurance function model behaves close to the experimental results, with loading conditions having $R = -1$, which conforms with the results reported in this paper in similar loading conditions. In the out of phase case for both materials, the results are in good agreement in the low cycle region but become more conservative as we move towards the high cycle region. One reason for this is that a greater number of experimental data points are in the low cycle region, which causes the calibration of the model to be biased towards the low cycle region. Also, there is scatter in the experimental data [9, 13], as is visible in Figures 5(b) and 6(b), which can lead to error in the prediction of fatigue life.

One simple solution to improve the prediction accuracy is to increase the number of calibration points, which will lead to better capture of the change in the experimental S-N curve trend, as shown in Figure 7. Here, one more calibration point is included with loading condition normal stress σ_a and shear stress $\tau_a = 140$ MPa with $R = -1$ and phase = 90° for C40 specimen (Table 1), which clearly resulted in improved agreement with the experimental S-N curve. So, it can be deduced from this that the data set required for

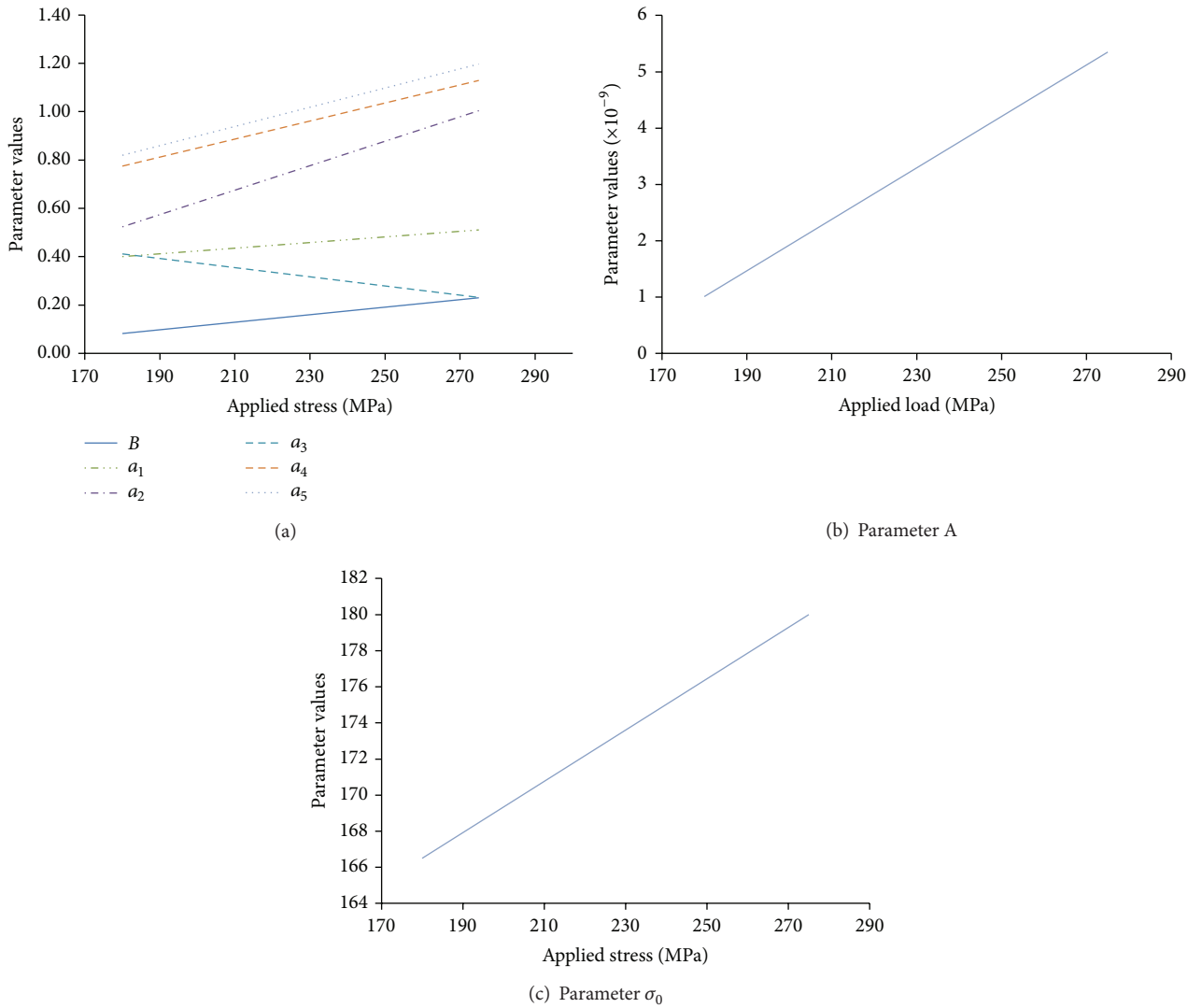
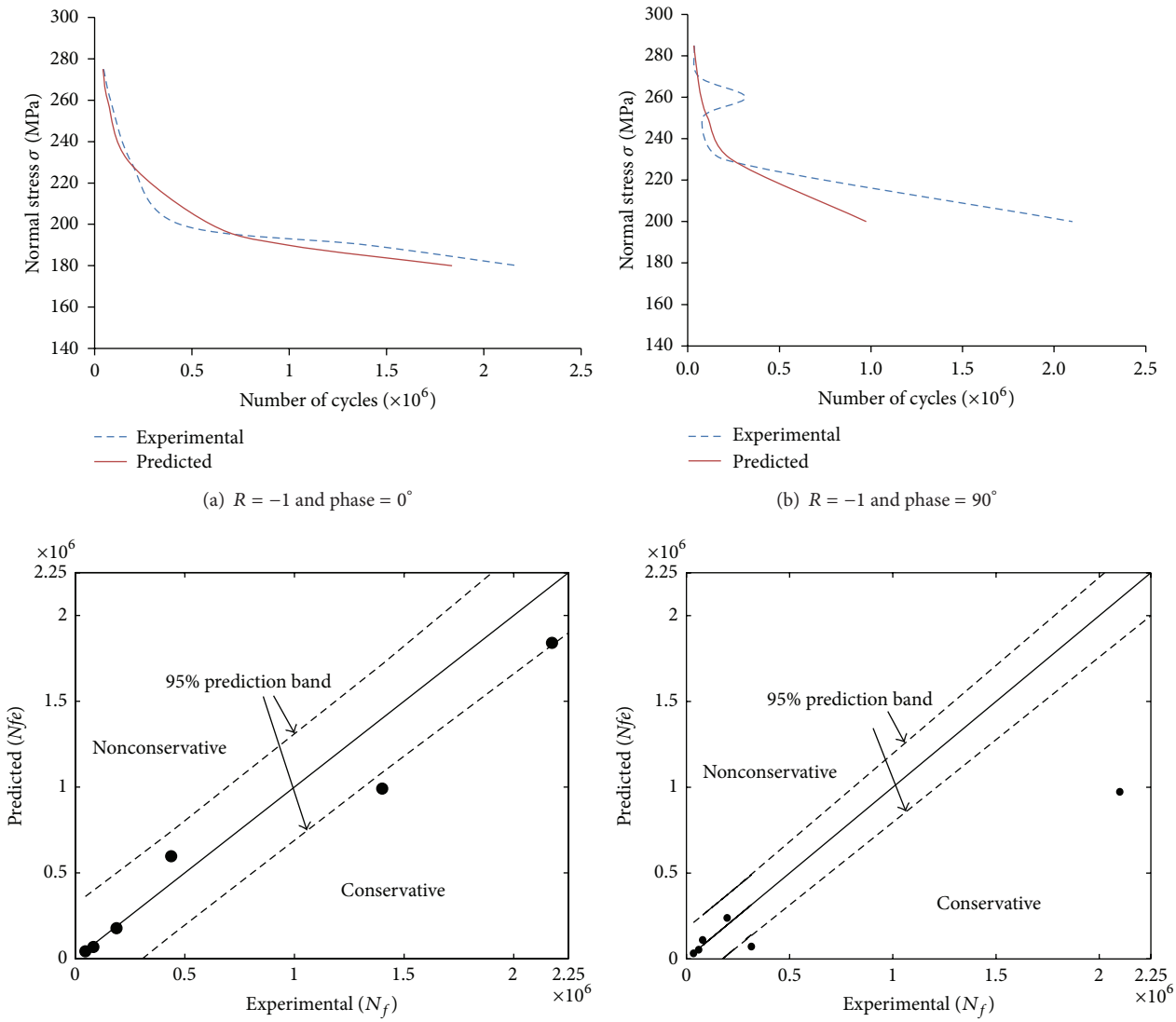


Figure 4: Trend of coefficients for calibration points for EN3B $R = -1$ phase = 0°.

the endurance model to work properly should have more data points and in both the low and high cycle regions, so that more calibration points can be used to fit the endurance model with the experimental data. The data set used in this study is small in size, which is why only one extra calibration point is included to check this hypothesis, and the results show that the idea of more calibration points will indeed work well. So, from these results, we can conclude that the methodology defined in this study can lead us to a model which works well in both the low and high cycle regions.

6. Conclusion

A methodology has been proposed to apply the endurance function model with a GA to estimate fatigue life. The proposed methodology included application of FEA, which in turn simplified the endurance function model by reducing one coefficient for the notch gradient correction factor. Also, application of FEA resulted in a simplified method for determining the stress tensor and deviatoric stress invariants. An interpolation technique is introduced in the proposed methodology to estimate the coefficients at each load point using calibration point coefficients, which resulted in better representation of the fatigue behavior from the endurance function model. The results show that the endurance function model is in good agreement with experimental fatigue life data for in-phase loading and also to some extent for out of phase loading, but, due to having fewer data points in the experimental data, the high cycle region is not accurately predicted. The scheme to use more than two calibration points for one data set shows improved prediction of the fatigue life. From the study, it is concluded that the proposed methodology for the endurance function model resulted in accurate prediction of fatigue life for the stable fatigue life behavior of the material. However, further development is needed to accommodate the nonlinear behavior of



(a) $R = -1$ and phase = 0°

(b) $R = -1$ and phase = 90°

(c) $R = -1$ and phase = 0°

(d) $R = -1$ and phase = 90°

Figure 5: S-N curve and experimental versus predicted life for EN3B steel.

the material in order to extend the application region of the endurance function model with the proposed methodology to include variable amplitude and random loading cases.

Nomenclature

D:	Fatigue damage
E:	Endurance function parameter
dD:	Damage increment
σ_a:	Normal stress amplitude at net area (MPa)
τ_a:	Shear stress amplitude at net area (MPa)
R:	Min stress/max stress = stress ratio
N_f:	Number of cycles to failure (experimentally)
$N_f e$:	Number of cycles to failure (predicted)
σ:	Stress tensor (MPa)
S_e:	Effective deviatoric stress tensor = $S - S_b$
S:	Applied deviatoric stress tensor
S_b:	Back stress tensor
$a_1, a_2, a_3, a_4, a_5, \sigma_0, A, B, V, C, h$:	Material constant parameters
dE:	Endurance function increment
G:	Notch gradient correction factor
γ:	Stress field parameter
dS_b:	Increment in back stress
e:	Error of predicted damage D
I_1, I_2, I_3:	Stress tensor invariants
$\sigma_1, \sigma_2, \sigma_3$:	Principal stresses
J_2, J_3:	Deviatoric stress invariants.

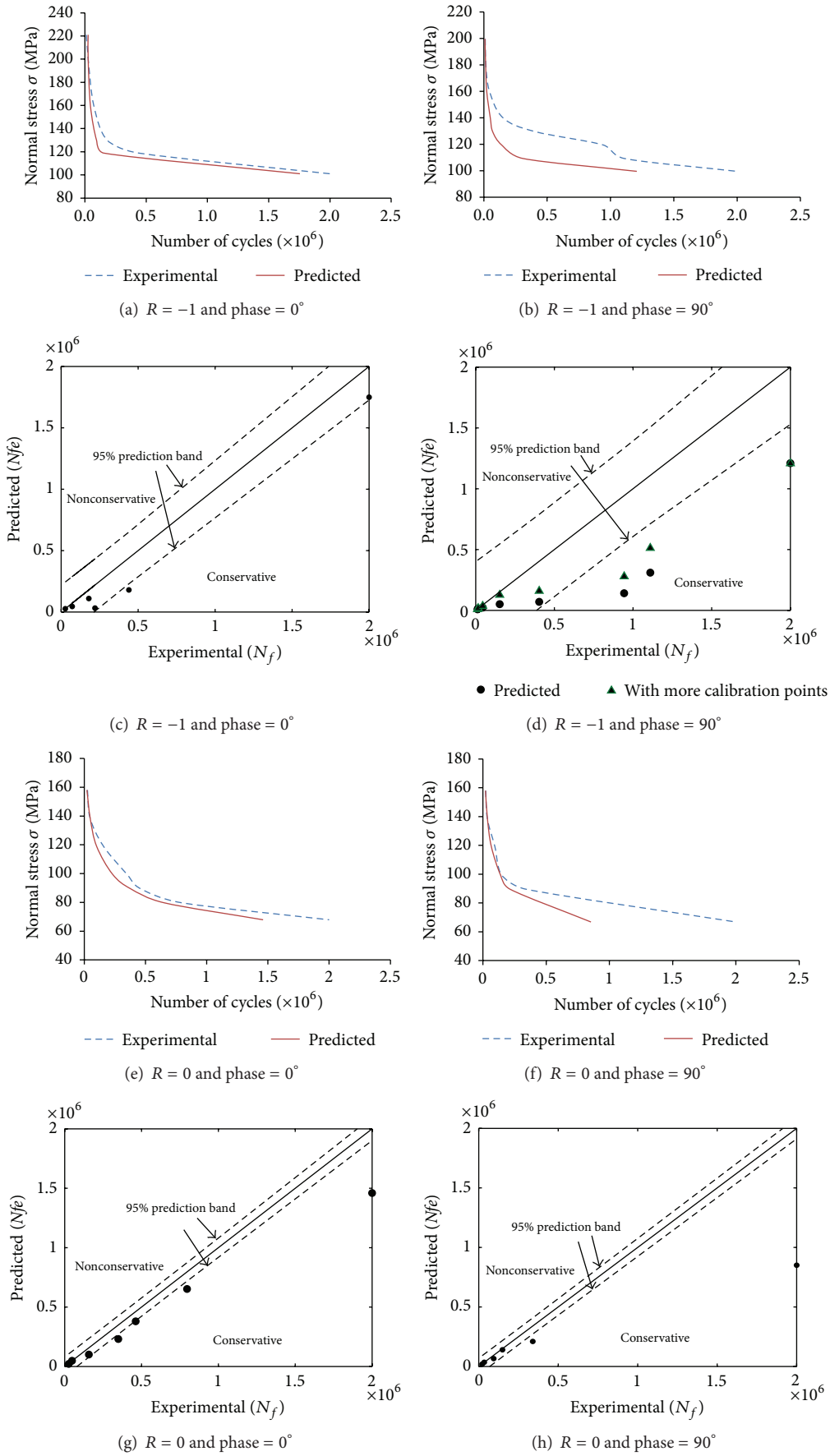


FIGURE 6: *S-N* curve and experimental versus predicted life for C40 steel.

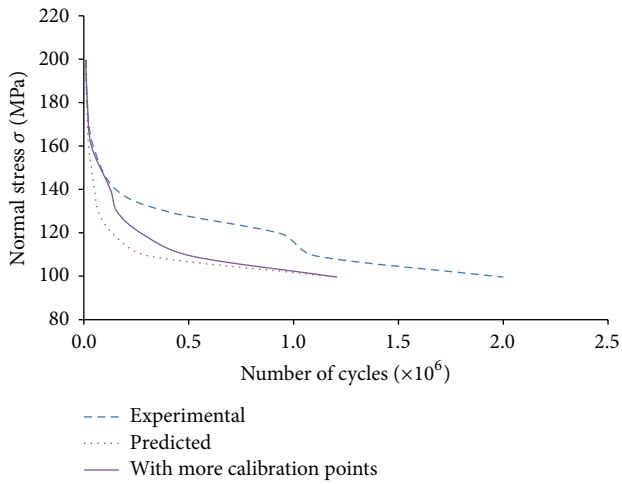


Figure 7: *S-N* curves for C40 steel for $R = -1$ and phase 90° with predicted curve from one more calibration point.

References

[1] Y. Lee, J. Pan, R. Hathaway, and M. Barkey, *Fatigue Testing and Analysis: Theory and Practice*, Butterworth Heinemann, New York, NY, USA, 2005.

[2] R. I. Stephens, A. Fatemi, R. R. Stephens, and H. O. Fuchs, *Metal Fatigue in Engineering*, John Wiley & Sons, New York, NY, USA, 2000.

[3] S. S. Manson and G. R. Halford, *Fatigue and Durability of Structural Materials*, ASM International, Materials Park, Ohio, USA, 2006.

[4] M. M. Rahman, A. K. Ariffin, N. Jamaludin, and C. H. C. Haron, "Influence of surface treatments on fatigue life of a two-stroke free piston linear engine component using random loading," *Journal of Zhejiang University Science A*, vol. 7, no. 11, pp. 1819–1830, 2006.

[5] M. M. Rahman, A. K. Ariffin, N. Jamaludin, and C. H. C. Haron, "Finite element based durability assessment of a free piston linear engine component," *Structural Durability and Health Monitoring*, vol. 3, no. 1, pp. 1–13, 2007.

[6] M. M. Rahman, K. Kadirgama, M. M. Noor, M. R. M. Rejab, and S. A. Kesulai, "Fatigue life prediction of lower suspension arm using strain-life approach," *European Journal of Scientific Research*, vol. 30, no. 3, pp. 437–450, 2009.

[7] M. M. Rahman, A. K. Ariffin, M. R. M. Rejab, K. Kadirgama, and M. M. Noor, "Multiaxial fatigue behavior of cylinder head for a free piston linear engine," *Journal of Applied Sciences*, vol. 9, no. 15, pp. 2725–2734, 2009.

[8] R. Brighenti and A. Carpinteri, "A notch multiaxial-fatigue approach based on damage mechanics," *International Journal of Fatigue*, vol. 39, pp. 122–133, 2012.

[9] L. Susmel and D. Taylor, "The modified Wöhler curve method applied along with the theory of critical distances to estimate finite life of notched components subjected to complex multiaxial loading paths," *Fatigue & Fracture of Engineering Materials & Structures*, vol. 31, no. 12, pp. 1047–1064, 2008.

[10] E. Macha and A. Niesłony, "Critical plane fatigue life models of materials and structures under multiaxial stationary random loading: the state-of-the-art in Opole Research Centre CESTI and directions of future activities," *International Journal of Fatigue*, vol. 39, pp. 95–102, 2012.

[11] L. Susmel and R. Tovo, "Estimating fatigue damage under variable amplitude multiaxial fatigue loading," *Fatigue & Fracture of Engineering Materials & Structures*, vol. 34, no. 12, pp. 1053–1077, 2011.

[12] L. Susmel and D. Taylor, "A critical distance/plane method to estimate finite life of notched components under variable amplitude uniaxial/multiaxial fatigue loading," *International Journal of Fatigue*, vol. 38, pp. 7–24, 2012.

[13] B. Atzori, F. Berto, P. Lazzarin, and M. Quaresimin, "Multiaxial fatigue behaviour of a severely notched carbon steel," *International Journal of Fatigue*, vol. 28, no. 5-6, pp. 485–493, 2006.

[14] N. W. M. Bishop and F. Sherratt, *Finite Element Based Fatigue Calculations*, NAFEMS, 2000.

[15] M. M. Rahman, A. K. Ariffin, S. Abdullah, M. M. Noor, and A. B. Rosli, "Durability assessment of cylinder block for two stroke free piston linear engine using random loading," *American Journal of Applied Sciences*, vol. 6, no. 4, pp. 726–735, 2009.

[16] M. M. Rahman, A. K. Ariffin, N. Jamaludin, S. Abdullah, and M. M. Noor, "Finite element based fatigue life prediction of a new free piston engine mounting," *Journal of Applied Sciences*, vol. 8, no. 9, pp. 1612–1621, 2008.

[17] M. M. Rahman, A. K. Ariffin, S. Abdullah, M. M. Noor, R. A. Bakar, and M. A. Maleque, "Finite element based fatigue life prediction of cylinder head for two-stroke linear engine using stress-life approach," *Journal of Applied Sciences*, vol. 8, no. 19, pp. 3316–3327, 2008.

[18] N. S. Ottosen, R. Stenström, and M. Ristinmaa, "Continuum approach to high-cycle fatigue modeling," *International Journal of Fatigue*, vol. 30, no. 6, pp. 996–1006, 2008.

[19] M. Kamal, M. M. Rahman, and A. G. A. Rahman, "Fatigue life evaluation of suspension knuckle using multi body simulation technique," *Journal of Mechanical Engineering and Sciences*, vol. 3, no. 1, pp. 291–300, 2012.

[20] R. Brighenti, A. Carpinteri, and S. Vantadori, "A genetic algorithm applied to optimisation of patch repairs for cracked plates," *Computer Methods in Applied Mechanics and Engineering*, vol. 196, no. 1–3, pp. 466–475, 2006.

[21] L. Davis, Ed., *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, NY, USA, 1991.

[22] M. Franulović, R. Basan, and I. Prebil, "Genetic algorithm in material model parameters' identification for low-cycle fatigue," *Computational Materials Science*, vol. 45, no. 2, pp. 505–510, 2009.

[23] V. B. Gantovnik, C. M. Anderson-Cook, Z. Gürdal, and L. T. Watson, "A genetic algorithm with memory for mixed discrete-continuous design optimization," *Computers and Structures*, vol. 81, no. 20, pp. 2003–2009, 2003.

[24] M. M. Rahman, A. K. Ariffin, M. R. M. Rejab, K. Kadirgama, and M. M. Noor, "Multiaxial fatigue behavior of cylinder head for a free piston linear engine," *Journal of Applied Sciences*, vol. 9, no. 15, pp. 2725–2734, 2009.

[25] M. M. Rahman, K. Kadirgama, M. M. Noor, M. R. M. Rejab, and S. A. Kesulai, "Fatigue life prediction of lower suspension arm using strain-life approach," *European Journal of Scientific Research*, vol. 30, no. 3, pp. 437–450, 2009.

[26] Society of Automotive Engineers (SAE), "Low cycle fatigue properties ferrous and non-ferrous materials," Tech. Rep. SAE J1099, Society of Automotive Engineers, 2002.

[27] D. F. Socie and G. B. Marquis, *Multiaxial Fatigue*, Society of Automotive Engineers, Warrendale, Pa, USA, 2000.

[28] A. Clarich, R. Russo, and M. Carriglio, "Multi-objective optimization with modeFRONTIER interfaces for ANSA and μETA Post," in *Proceedings of the 4th ANSA & μETA International Conference*, Thessaloniki, Greece, 2011.

[29] M. Perillo, V. Primavera, L. Fuligno, G. Fabbri, C. Steenbergen, and N. Pasini, "Optimization and robustness of complex material model simulations with modeFRONTIER," in *Proceedings of the 7th European LS-DYNA Conference*, pp. 1–10, Salzburg, Austria, 2009.

[30] *modeFRONTIER 4.1 User Manual*, ESTECO, Trieste, Italy, http://www.esteco.com/modefrontier.

[31] K. Hariharan, R. V. Prakash, and M. Sathya Prasad, "Weighted error criterion to evaluate strain-fatigue life prediction methods," *International Journal of Fatigue*, vol. 33, no. 5, pp. 727–734, 2011.

[32] R. Brighenti, A. Carpinteri, and N. Corbari, "Damage mechanics and Paris regime in fatigue life assessment of metals," *International Journal of Pressure Vessels and Piping*, vol. 104, pp. 57–68, 2013.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 760126, 10 pages
http://dx.doi.org/10.1155/2013/760126

Research Article

Path Planning for Detection Robot Climbing on Rotor Blade Surfaces of Wind Turbine Based on Neural Network

Binrui Wang,[1,2] Haohua Luo,[1] Yinglian Jin,[1] and Mewei He[1]

[1] *College of Mechanical and Electrical Engineering, China Jiliang University, Hangzhou 310018, China*
[2] *College of Engineering, University of Tennessee, Knoxville, TN 37996, USA*

Correspondence should be addressed to Binrui Wang; wangbinrui@163.com

Received 8 August 2013; Accepted 4 October 2013

Academic Editor: Shengyong Chen

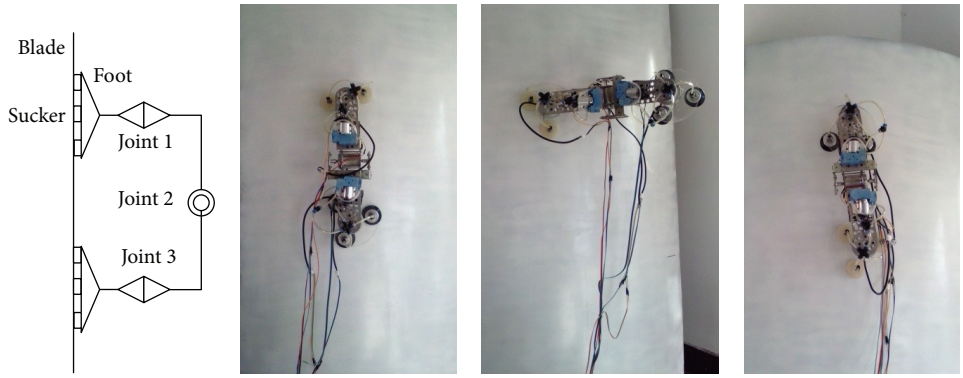
Two-feet climbing robot is proposed for rotor blade surface damage detection. The penalty function is designed based on the simulated annealing neural network for waypoints inside blade. According to the derivative of path energy function to time, waypoints are updated and move toward the direction reducing the path energy consisting of length and penalty function. According to the curvature variation range, a novel weighted simulated-annealing-temperature updating method is designed to get comprehensive optimization of the path energy and convergence speed. The path planning is accomplished for the root, middle, and tip blade parts, respectively. The asymptotic analysis of the waypoint coordinating updating process was given, and the updated start point is adopted during escaping from inside. The effect of weight on path energy and convergence speed is analyzed. The planning results show the effectiveness of the proposed path planning algorithm, and the weighted average method is valid for the comprehensive optimization.

1. Introduction

Bearing the huge wind stress during energy conversion process, the rotor blade is among the most easy-to-failure components of wind turbine generator system (WTGS) [1]. Due to that blades are vulnerable to cumulative fatigue damage owing to the periodic nature of the loading, detecting damage to blades before failure is crucial. Currently, rotor blades damage is only detected manually, which is dangerous and low effective. The authors propose to use the climbing robot to detect blades surfaces damage.

Aiming to find an optimal path from start to target point in an unstructured environment, the path planning is one of the fundamental problems in robots detecting surface automatically. The path planning can be classified into two major categories, global and local path planning, according to the environment information [2–4]. The global path planning creates a path in a static environment, while the local path planning does in a dynamic environment. Due to the dynamic obstacles, the obstacle avoidance method is the key in the local path planning, such as the virtual force field, Elman fuzzy adaptive control [5]. Owing to lack of complete environment information, local path planning algorithm is more complex than global path planning algorithm, but the latter requires the environment modeling and path searching. The literature shows that researchers have studied the visibility graphs method [6], free space method [7], topological method [8], and grid method [9] for environment modeling. These methods are mainly applied to two-dimension (2D) space.

We study the path planning for the three-dimension (3D) curved solid, and rotor blade mathematical expressed equation is more difficult. To detect blade surfaces, the climbing robot should move along the rotor blade surfaces. To this day, researchers have used the mixed integer linear programming algorithm (MILP), ant colony optimization and neural network (NN) to study the path planning method for the climbing robot in 3D space. The MILP-based path planning for city-climber robot is complex and the path length is not considered as a constraint condition [10]. Hu and Cai have presented an improved ant colony algorithm to study the mobile robot path planning in 3D space with the given environment information [11]. This algorithm's shortcoming is being time consuming and not suitable for



(a) (b)

Figure 1: The mechanism and snapshots of climbing robot on the wind turbine blade: (a) the mechansim sketch; (b) the snapshots.

the path planning on the surface of curved solids. Neural network is biologically inspired and robust to time-varying delay [12]. Yu et al. have proposed an aggressive algorithm based on the energy function using the artificial NN to describe the obstacles [13, 14], but this algorithm is not fit for the nonlinear boundary surfaces. Yang and Meng studied a dynamic collision-free trajectory generation using NN to avoid concave U-shaped obstacles [15]. For concave objects minimum distance problem, Carretero and Nahon designed the simulated annealing (SA) and genetic algorithms (GAs) [16]. Blade includes both the convex and concave; therefore, the path planning for the minimum distance is more complicated problem. For the stability of NN, the weighting method is valid, and Zhang et al. gave a solution based on optimization method to calculate the optimal weighting-delay parameters [17].

To the best of our knowledge, there still is not a suitable path planning for robot climbing on the blade surfaces of wind turbine. This paper studies a 3D space path planning method for rotor blade climbing robot based on the artificial neural network and simulated annealing described in [18]. The main contribution of this paper is to extend the NN path planning from linear boundary to nonlinear boundary surfaces including both convex and concave in 3D. Meanwhile, the weighted average of SA temperature is designed to improve the comprehensive performance of the planning speed and the path length. Furthermore, the weight formulation is designed according to the surface's curvature.

This paper is organized as follows. In Section 2, we firstly give the climbing robot prototype and analyze the curvature varying of blade. The penalty function between waypoints and 3D curved solids is designed based on artificial neural network. Path energy function is defined and waypoints moving directions are derived. The weighted calculation of the simulated annealing temperature is proposed for the comprehensive optimization of path energy and convergence speed. In Section 3, we give the detailed parameters of path planning. Simulations on whole blade are conducted and the annealing temperature weight adjustment is done to illustrate the significance of proposed weight formulation. Computing results are shown at the end of Section 3. The conclusion is drawn in Section 4.

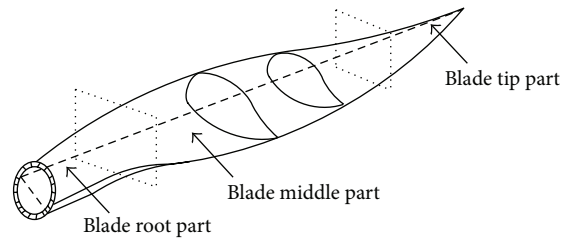


Figure 2: Division of rotor blade.

2. Path Planning Design Using Neural Network

2.1. Climbing Robot Prototype and Rotor Blade Curvature. We have developed a climbing robot for blade damage detection shown in Figure 1. We combine climbing techniques with a modular approach to realize a practical climbing robotic based on an easy-to-build mechanical structure [19]. This robot has two feet and three degrees of freedom, and each foot has three vacuum suckers installed in the vertex of a triangle. All joints are actuated by DC servomotor. The robot mechanical structure consists of foot modules (Joint 1 and Joint 2) and body module (Joint 3). Through adding the foot or body module, we can get the more flexible caterpillar-climbing robot expediently. This robot can climb rotor blade surfaces with swing-around gait and carry the camera, ultrasonic or infrared instrument to detect the blade.

This two-feet climbing robot has light weight and only two interaction contact areas with blade surface, so it can be more easily adaptive to varying curves than climbing robot with more feet. In path planning, the robot can be simplified as the two-link mechanism. The Joint 2 at the center of mass or shape connects two links, and the end of each link is the foot. The planned path is composed of a number of waypoints between start and target point. These waypoints mean the continuous footprints.

It is difficult to express the whole rotor blade accurately by mathematical equations, so we divide the rotor blades into three parts according to the curvature, as shown in Figure 2.

Table 1: Parameters of the rotor blade.

Part	Chord length (m)	Curvature variation range (m^{-1})	Number of surfaces
Root	2	0.5	3
Middle	1.68–3.03	0.67–4	4
Tip	0.64–1.68	0.11–3.00	2

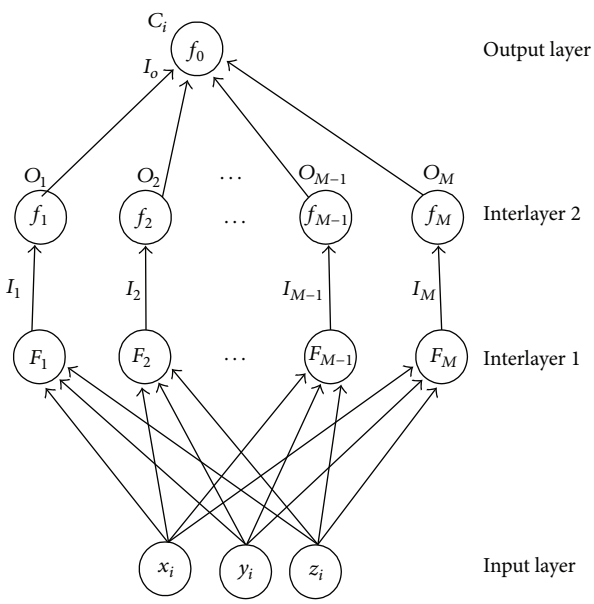


Figure 3: Artificial neural network of a waypoint's penalty function.

According to [20], we properly idealize the curvature variation range of each part shown in Table 1. The curvature variation range of root part is very small, so we model this part as a cylinder. The curvature variation range of middle part is relatively large, so we model this part as a polyhedron surrounded by regular curved surfaces. The tip part is modeled as an elliptic cone.

For climbing motion, footprints should be on the blade surface proper, so we should punish these waypoints under or out of blade surface.

2.2. Penalty Function of Point Based on Artificial Neural Network. Penalty function method is usually used to solve the constrained minimization problems. The key point of this method is to obtain an augmented objective function by adding a penalty function to the original objective function. Namely, penalty function should add a great penalty value to these points attempting to cross the border or escape from the feasible region. In this paper, we define the path penalty function as the sum of all waypoints' penalty function. The value of each waypoint penalty function is obtained using the artificial neural network shown in Figure 3.

Three nodes in input layer denote the x, y, and z coordinate values of the given waypoint in the Cartesian. The boundary surface is defined as $F_m(x, y, z) = 0$, and then constraint equations of the given blade curved solid can be shown as

$$F_m(x, y, z) > 0, \quad m = 1, 2, \ldots, M, \tag{1}$$

where M denotes the count of boundary surfaces of the given blade sector.

In the Interlayer 1 and Interlayer 2, the count of nodes is equal to M. Moreover, the value of node connection weight between input layer, interlayer, and output layer is set to 1 specially.

Input and output of above artificial neural network nodes are

$$\begin{aligned} I_m &= F_m(x_i, y_i, z_i), \\ O_m &= f_m(I_m), \\ I_o &= \sum_{m=1}^{M} O_m + \theta_I, \end{aligned} \tag{2}$$

$$C_i = f_0(I_o), \quad i = 2, 3, \ldots, N-1, \tag{3}$$

where I_m and O_m are the input and output of node m in the Interlayer 2, respectively; I_o is the input of the node in the output layer; C_i is the output of the node in the output layer, namely, the penalty function value of waypoint $P_i(x_i, y_i, z_i)$; θ_I is the threshold of the node in output layer; N is the count of discrete waypoints that depends on the climbing step length and the sizes of robot mechanism. Excitation functions of the nodes in the output layer and Interlayer 2 adopt S type Sigmoid functions as follows:

$$\begin{aligned} f_0(x) &= \frac{1}{1 + e^{-x/T_o(t)}}, \\ f_m(x) &= \frac{1}{1 + e^{-x/T_m(t)}}, \end{aligned} \tag{4}$$

where $T_0(t)$ and $T_m(t)$ are the annealing temperature varying with time t.

Define the value of the whole path penalty function as

$$E_c = \sum_{i=1}^{N} C_i, \tag{5}$$

where C_i is between 0 and 1, which reflects the collision risk between the waypoint and the curved solid.

2.3. Path Energy Function and Planning Algorithm. Path energy function is used to represent the energy consumption of robot climbing along the given path. We define the path energy as the sum of the path length and penalty function

$$E = E_l + E_c, \tag{6}$$

where E_l represents the sum of square of each segment length $L_i (L_0 = L_N = 0)$,

$$E_l = \sum_{i=1}^{N-1} \left((x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2 \right). \tag{7}$$

Waypoints should move towards the direction decreasing the path energy. Namely, waypoints should move from the inside to curved solid surfaces, and the path length should be restrained as short as possible.

We formulate the derivative of the energy function to time and get

$$\begin{aligned}\dot{E} = &\sum_{i=1}^{N}\left[\left(\frac{\partial L_i^2}{\partial x_i}+\frac{\partial L_{i-1}^2}{\partial x_i}\right)+\frac{\partial C_i}{\partial x_i}\right]\dot{x}_i \\ &+\sum_{i=1}^{N}\left[\left(\frac{\partial L_i^2}{\partial y_i}+\frac{\partial L_{i-1}^2}{\partial y_i}\right)+\frac{\partial C_i}{\partial y_i}\right]\dot{y}_i \\ &+\sum_{i=1}^{N}\left[\left(\frac{\partial L_i^2}{\partial z_i}+\frac{\partial L_{i-1}^2}{\partial z_i}\right)+\frac{\partial C_i}{\partial z_i}\right]\dot{z}_i.\end{aligned} \tag{8}$$

Define

$$\begin{aligned}\dot{x}_i &= -\eta_1\left[\left(\frac{\partial L_i^2}{\partial x_i}+\frac{\partial L_{i-1}^2}{\partial x_i}\right)+\frac{\partial C_i}{\partial x_i}\right], \\ \dot{y}_i &= -\eta_1\left[\left(\frac{\partial L_i^2}{\partial y_i}+\frac{\partial L_{i-1}^2}{\partial y_i}\right)+\frac{\partial C_i}{\partial y_i}\right], \\ \dot{z}_i &= -\eta_1\left[\left(\frac{\partial L_i^2}{\partial z_i}+\frac{\partial L_{i-1}^2}{\partial z_i}\right)+\frac{\partial C_i}{\partial z_i}\right],\end{aligned} \tag{9}$$

where η_1 is a positive constant that denotes the optimizing rate of energy function.

Substituting equation set (9) into (8), we can get

$$\dot{E} = -\frac{1}{\eta_1}\sum_{i=1}^{N}\left[(\dot{x}_i)^2+(\dot{y}_i)^2+(\dot{z}_i)^2\right] < 0. \tag{10}$$

So equation set (9) can ensure that waypoints are moving towards the direction decreasing the path energy.

Then by substituting (3) and (7) into equation set (9), we get the dynamic planning equation of waypoint $P_i(x_i, y_i, z_i)$ as follows:

$$\begin{aligned}\dot{x}_i = &-2\eta_1(2x_i - x_{i-1} - x_{i+1}) \\ &-\eta_1\frac{1}{T(t)}f_0(I_o)[1-f_0(I_o)] \\ &\times\sum_{m=1}^{M}\left\{\frac{1}{T_m(t)}f_m(I_m)[1-f_m(I_m)]\frac{\partial I_m}{\partial x_i}\right\}, \\ \dot{y}_i = &-2\eta_1(2y_i - y_{i-1} - y_{i+1}) \\ &-\eta_1\frac{1}{T(t)}f_0(I_o)[1-f_0(I_o)] \\ &\times\sum_{m=1}^{M}\left\{\frac{1}{T_m(t)}f_m(I_m)[1-f_m(I_m)]\frac{\partial I_m}{\partial y_i}\right\}, \\ \dot{z}_i = &-2\eta_1(2z_i - z_{i-1} - z_{i+1}) \\ &-\eta_1\frac{1}{T(t)}f_0(I_o)[1-f_0(I_o)] \\ &\times\sum_{m=1}^{M}\left\{\frac{1}{T_m(t)}f_m(I_m)[1-f_m(I_m)]\frac{\partial I_m}{\partial z_i}\right\}.\end{aligned} \tag{11}$$

The previous section of equation set (11) is used for optimizing the path length, while the latter section is used for making waypoints to move from curved solid inside to surfaces. So waypoints within the curved solid should move according to equation set (11).

For waypoints on the curved solid surfaces, the latter section of equation set (11) is unnecessary, so these waypoints should move according to the equations as follows:

$$\begin{aligned}\dot{x}_i &= -2\eta_2(2x_i - x_{i-1} - x_{i+1}), \\ \dot{y}_i &= -2\eta_2(2y_i - y_{i-1} - y_{i+1}), \\ \dot{z}_i &= -2\eta_2(2z_i - z_{i-1} - z_{i+1}),\end{aligned} \tag{12}$$

where η_2 is a positive constant denoting the optimizing rate of path length.

Equation set (11) and (12) indicate the rough and careful adjustment of a waypoint location, respectively. So we usually set $\eta_1 > \eta_2$ in the path planning algorithm.

2.4. Temperature Updating Using Simulated Annealing. The energy function above may have multiple extremum values, as a result, and the path planning algorithm may fall into local minimum points quickly. In order to escape from the local minimum problem as far as possible, we applied simulated annealing method to change the value of $T_o(t)$ and $T_m(t)$. Due to that the climbing robot must cost energy to adhere to the blade even in static state, the planning time and path length are coequally important. The global minimum without considering convergence speed is not practical.

Table 2 summarizes and compares the existing temperature updating functions about simulated annealing methods in the literature. Table 2 compares three main performances including the probability of escaping from local minima, the convergence speed, and the correlation with variable dimension of optimization objective.

For online path planning, we want to design a temperature updating function that can escape from local minimum as far as possible and enhance convergence speed to reduce the planning time simultaneously.

Table 2: Performances comparison of different temperature updating functions.

Source of algorithms	Temperature updating function	Probability of escaping from local minima	Convergence speed	Correlation with variable dimension of optimization objective
Reference [21]	$T_{o(m)}(t) = \dfrac{\beta_{o(m)}}{\log(1+t)}$	Must be able to escape	Relatively slow	Uncorrelated
Reference [22]	$T_{o(m)}(t) = \dfrac{\beta_{o(m)}}{1+t}$	Might not be able to escape	Very fast	Uncorrelated
Reference [23]	$T_{o(m)}(t) = \dfrac{\beta_{o(m)}}{\log(t^{1/n})} \quad (n \geq 1)$	Must be able to escape	Slow	Correlated
Reference [23]	$T_{o(m)}(t) = \dfrac{\beta_{o(m)}}{t^{1/n}} \quad (n \geq 1)$	Might not be able to escape	Fast	Correlated
Reference [24]	$T_{o(m)}(t) = \dfrac{\beta_{o(m)}}{\log(t^{q/n})} \quad (n > 1, 1 \leq q \leq n)$	Must be able to escape	Depend on q	Correlated
Reference [25]	$T_{o(m)}(t) = \dfrac{\beta_{o(m)}}{t^m} \quad (m = 1, 2, \ldots)$	Might not be able to escape	Fast	Uncorrelated

Design $T_o(t)$ and $T_m(t)$ as follows:

$$T_o(t) = \alpha \frac{\beta_o}{\log(1+t)} + (1-\alpha)\frac{\beta_o}{1+t} \quad 0 < \alpha \leq 1,$$
$$T_m(t) = \alpha \frac{\beta_m}{\log(1+t)} + (1-\alpha)\frac{\beta_m}{1+t} \quad 0 < \alpha \leq 1, \tag{13}$$

where α is the weight, and β_0 and β_m are the initial simulated annealing temperature in the output layer and Interlayer 2, respectively.

Curvature of boundary surfaces is not been considered in the past literature. Intuitively, the path planning is easy to fall into the local minimum due to large varying range of curvature, so in this case we should increase α to reduce the speed of temperature change. Therefore, in flat surface, α should be less for quick convergence, while in curve surface, α should be larger for escaping from local minima. Therefore, the surface curvature variation is the key to setting α value. We creatively design α as the following formulation:

$$\alpha = \begin{cases} \dfrac{1}{1+0.65e^{k_{\min}/k_{\max}}} & M = 1 \\ \dfrac{1}{1+0.65e^{\min\{k_{m\max}\}/\max\{k_{m\min}\}}} & M > 1, \end{cases} \tag{14}$$

where $k_{m\min}$ and $k_{m\max}$ denote the minimum and maximum curvature of the m boundary surface, respectively.

2.5. Path Planning Processing. We assume that start point $P(x_1, y_1, z_1)$ and target point $P(x_N, y_N, z_N)$ are located on the curved solid surface. Design the path planning processing as follows.

(1) *Step One.* The initial path is composed of evenly distributed waypoints along the straight line between start and target point, and the waypoints coordination equations are

$$x_i = x_1 + \frac{i(x_N - x_1)}{(N-1)},$$
$$y_i = \frac{(y_N - y_1)(x_i - x_1)}{(x_N - x_1)} + y_1,$$
$$z_i = \frac{(z_N - z_1)(x_i - x_1)}{(x_N - x_1)} + z_1,$$
$$i = 2, 3, \ldots, N-1, \ x_1 \neq x_N. \tag{15}$$

(2) *Step Two.* If $I_m > 0$ $(m = 2, 3, \ldots, M)$, then waypoint $P(x_i, y_i, z_i)$ is inside the curved solid and should move according to equation set (11); otherwise, $P(x_i, y_i, z_i)$ is outside the curved solid and should move according to Equation set (12). Repeat Step two until all waypoints are outside the solid.

(3) *Step Three.* Repeat moving all waypoints according equation set (12) until the path energy function *converges* to the minimum. In this paper, converge condition is defined as the changing of the path energy value is always less than 10^{-5} in ten cycles continuously.

Figure 4 shows the algorithm flow chart.

3. Path Planning Experiments

3.1. Planning Condition. For validating the effect of the proposed path planning algorithm was implemented using MATLAB2009b on a personal computer (PC) with Intel(R) Core(TM) 2 Duo T5450 @ 1.66 GHz CPU. The Runge-Kutta method was adopted for solving equation set (11) and (12).

The rotor blade for test is shown in Figure 2. For practical computing, we established simply the constraint equations of each part based on Table 1. Constraint equations of boundary

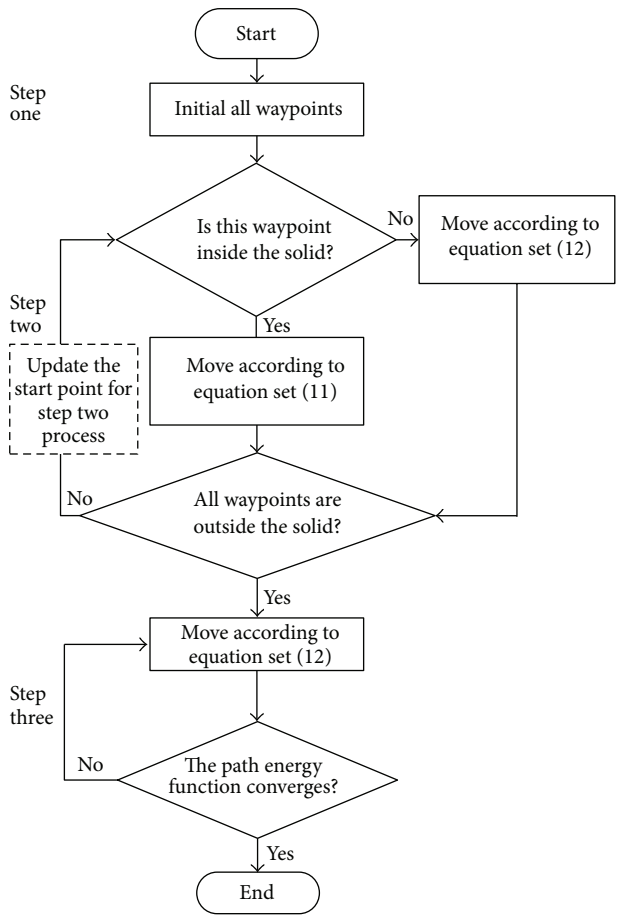


Figure 4: Flowchart of path planning algorithm.

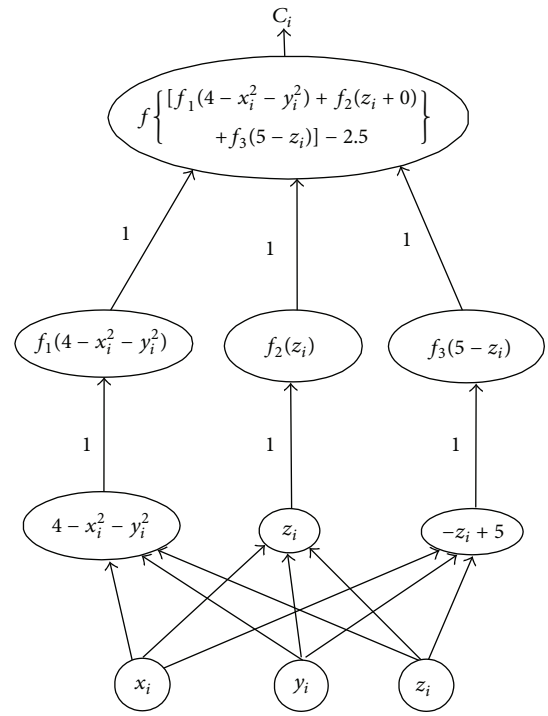


Figure 5: Neural network between the given waypoint and blade root part.

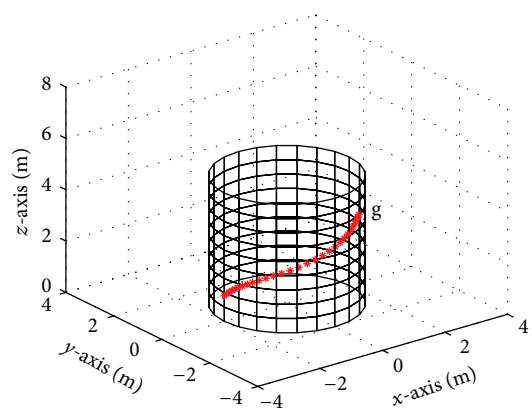


Figure 6: Path planning result on the surface of blade root part.

surfaces for the blade root, middle, and tip parts are shown in (16) successively. Consider

$$\begin{gathered} F_1(x,y,z) = 4 - x^2 - y^2 > 0, \\ F_2(x,y,z) = z > 0, \\ F_3(x,y,z) = 5 - z > 0, \\ F_1(x,y,z) = 5 - z + \sin 2y > 0, \\ F_2(x,y,z) = 2.25 - (y-3.5)^2 - (z-4.456)^2 > 0, \\ F_3(x,y,z) = x > 0, \\ F_4(x,y,z) = -x + 5 > 0, \\ F_1(x,y,z) = z^2 - 9x^2 - y^2 > 0, \\ F_2(x,y,z) = 3 - z > 0. \end{gathered} \tag{16}$$

3.2. Path Planning Simulation Study. Planning experiments for each part were conducted.

Figure 5 presents the neural network of the penalty function between waypoint $P(x_i, y_i, z_i)$ and blade root part. Figure 6 gives the path planning result on the boundary surfaces of root part. In Figure 6, "s" means the start point, and "g" means the target point.

The value of α is set to 0.36 according to the curvature variation range in Table 1 and (14). Other parameters are as follows:

$$\begin{gathered} N = 30, \quad \beta_0 = 0.5, \quad \beta_m = (0.5, 0.6, 0.6), \\ \eta_1 = 0.90, \quad \eta_2 = 0.25. \end{gathered} \tag{17}$$

For path planning on the boundary surfaces of middle part, the simulation parameters are as follows:

$$\begin{gathered} N = 40, \quad \beta_0 = 0.5, \quad \beta_m = (0.6, 0.6, 0.5, 0.5), \\ \eta_1 = 1.30, \quad \eta_2 = 0.50. \end{gathered} \tag{18}$$

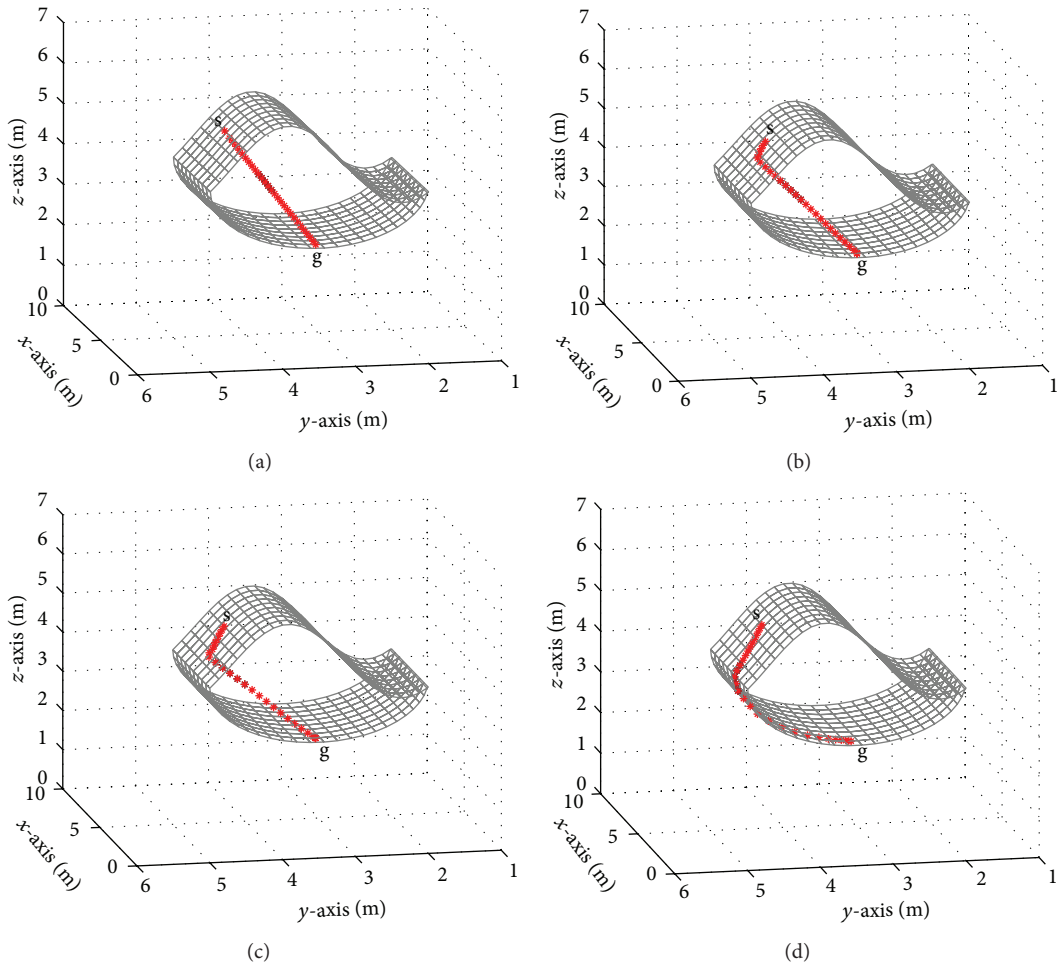


Figure 7: Path planning processing of the algorithm on middle part: (a) initial state; (b) $K = 35$; (c) $K = 40$; (d) $K = 71$.

The value of α is set to 0.57 according to the curvature variation range in Table 1 and (14).

Figure 7 presents the process of path planning on the blade middle part. K denotes the number of the algorithm loops.

Figure 7 shows that the proposed planning algorithm is conducted in turn on different boundary surfaces. Furthermore, we found that waypoints orderly escaped from curved solid inside to surfaces according to the sequence from start to goal point. Although, all waypoints on the path were updated on each iterative planning, from start to goal point. Therefore, this path planning is asymptotic and all waypoints on the path are updated on each iterative planning. Meanwhile, we found the coordinate varying is tiny and less than 0.01 m after waypoints arrived on surfaces. So we designed a quick planning algorithm for Step two to reduce the computing burden, through updating the new start point using the waypoint just moving to the surface. However, this updating is only for Step two. When the algorithm flows into Step three, the start point should be returned to original start point. The quick planning for Step two is shown in dotted line frame in Figure 4. Planning experiments show that the computing time is 4.07 s without and 2.81 s with updating the start point, respectively.

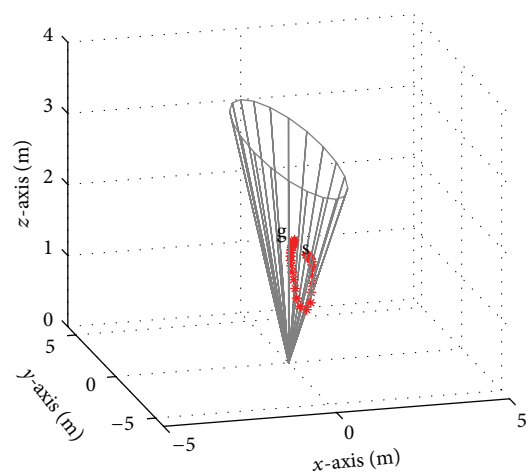


Figure 8: Path planning result on the surface of blade tip part.

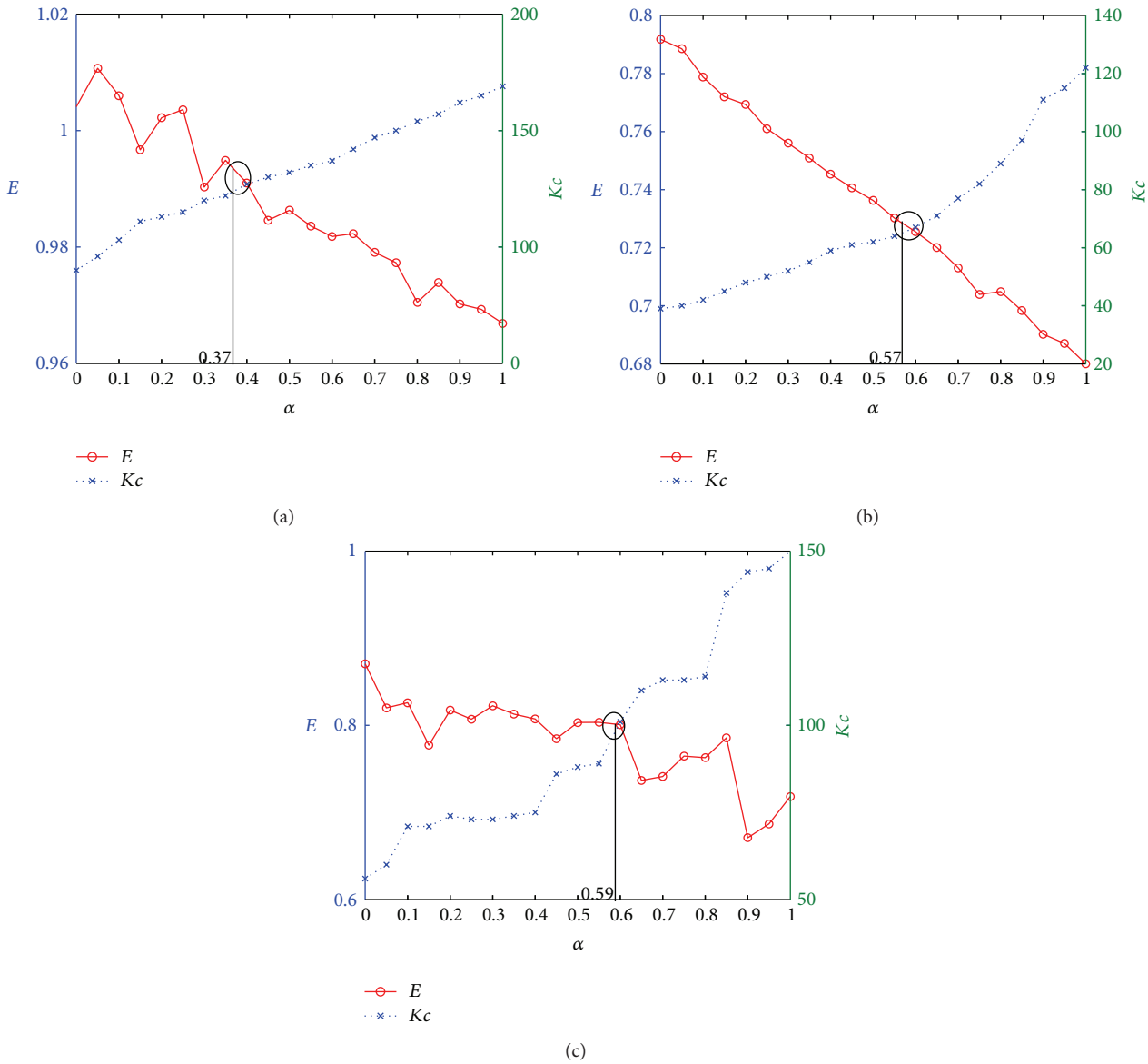


Figure 9: The relationships between the energy function and convergence speed with weights: (a) blade root part; (b) blade middle part; (c) blade tip part.

For path planning on the boundary surfaces of tip part, the simulation parameters are the following:

$$N = 20, \qquad \beta_0 = 0.5, \qquad \beta_m = (0.5, 0.5), \qquad \eta_1 = 1.10, \qquad \eta_2 = 0.25. \tag{19}$$

The value of α is set to 0.59 according to the curvature variation range in Table 1 and (14).

Figure 8 presents the path planning results on boundary surfaces of tip part.

Figures 6, 7, and 8 show the effect and success of the proposed algorithm on whole rotor blade boundary surfaces.

3.3. Annealing Temperature Weight Adjustment. Experiments on the whole blade with different α were conducted to demonstrate the performance of the proposed annealing temperature weight in (14) and (15). E and Kc denote the energy function value and convergence speed in Step two of algorithm flow, respectively.

Figure 9 shows the value of E and Kc varying with annealing temperature weight with step 0.05 in the range 0-1. The convergence speed Kc increases monotonically with increasing α in Figures 9(a) and 9(b), and nearly increases monotonically in Figure 9(c). However, the energy function E decreases non-monotonically with increasing α in all Figure 9. The value of α calculated from (14) is at or close to the comprehensive optimum point for the energy and convergence speed, namely, the intersection point of two curves.

Specially, the path planning in the elliptic cone surface easily escapes form the cone inside when start and goal point

are near the cone tip. Therefore, the effect of α to E is not distinct relatively.

The results have shown that the proposed temperature weight formulation is capable of synchronously improving the convergence speed and optimizing the path.

4. Conclusion

This paper presents a path planning algorithm for climbing robot on the rotor blade surfaces of wind turbine, based on neural network and simulated annealing method. According to path planning results, we can make the following conclusions.

(1) The proposed path planning algorithm can obtain an optimized feasible path along rotor blade boundary surface as short as possible successfully.

(2) Updating the start point can reduce the planning computing time due to the asymptotic path characteristics.

(3) The weighted average of temperature updating function is benefit to comprehensive optimum of the energy function and convergence speed. The weight formulation according to the curvature variation range of boundary surfaces is useful and simple.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by the Natural Science Foundation of Zhejiang province (LQ13E050004) and the Research Project of General Administration of Quality Supervision, Inspection and Quarantine of China (no. 201210076).

References

[1] X. J. Li, Z. M. Xue, and J. H. Wang, "Situation and development trend of large wind turbines blade design technique," *Glass Fiber Reinforced Plastics/Composite Material*, vol. 1, no. 1, pp. 48–52, 2008.

[2] C.-C. Tsai, H.-C. Huang, and C.-K. Chan, "Parallel elite genetic algorithm and its application to global path planning for autonomous robot navigation," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 10, pp. 4813–4821, 2011.

[3] C. C. Hsu, Y. J. Chen, M. C. Lu, and S. A. Li, "Optimal path planning incorporating global and local search for mobile robots," in *Proceedings of the 1st IEEE Global Conference on Consumer Electronics (GCCE '12)*, pp. 668–671, IEEE, Tokyo, Japan, 2012.

[4] T. M. Howard, C. J. Green, A. Kelly, and D. Ferguson, "State space sampling of feasible motions for high-performance mobile robot navigation in complex environments," *Journal of Field Robotics*, vol. 25, no. 6-7, pp. 325–345, 2008.

[5] S. Wen, W. Zheng, J. Zhu, X. Li, and S. Chen, "Elman fuzzy adaptive control for obstacle avoidance of mobile robots using hybrid force/position incorporation," *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 42, pp. 603–608, 2011.

[6] G. Dudek and M. Jenkin, *Computational Principles of Mobile Robotics*, Cambridge University Press, Cambridge, UK, 2000.

[7] T. Ducket and U. Nehmzow, "Experiments in evidence-based localization for a mobile robot," in *Proceeding of the AISB Workshop on Spatial Reasoning in Mobile Robots and Animals*, pp. 245–255, Manchester, UK, April, 1997.

[8] J. Kim, R. A. Pearce, and N. M. Amato, "Robust geometric-based localization in indoor environments using sonar range sensors," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS '02)*, pp. 421–426, October 2002.

[9] K. Kondo and F. Kimura, "Collision avoidance using a free space enumeration method based on grid expansion," *Advanced Robotics*, vol. 3, no. 3, pp. 159–175, 1989.

[10] R. Yue, J. Xiao, S. L. Joseph, and S. Wang, "Modeling and path planning of the city-climber robot part II: 3D path planning using mixed integer linear programming," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO '09)*, pp. 2391–2396, December 2009.

[11] H. Hu and X. S. Cai, "Path planning of robots in 3D space based on an improved ant colony algorithm," *Computer Systems and Applications*, vol. 20, no. 11, pp. 95–98, 2011.

[12] J. Fu, H. Zhang, T. Ma, and Q. Zhang, "On passivity analysis for stochastic neural networks with interval time-varying delay," *Neurocomputing*, vol. 73, no. 4–6, pp. 795–801, 2010.

[13] J.-L. Yu, S.-Y. Cheng, Z.-Q. Sun, and V. Kroumov, "An optimal algorithm of 3D path planning for mobile robots," *Journal of Central South University*, vol. 40, no. 2, pp. 471–477, 2009.

[14] J.-L. Yu, X.-M. Zhang, S.-Y. Cheng, Z.-Q. Sun, and V. Kroumov, "Approach on path planning for climbing robot," *Journal of System Simulation*, vol. 21, no. 15, pp. 4748–4751, 2009.

[15] S. X. Yang and M. Meng, "Neural network approaches to dynamic collision-free trajectory generation," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 3, pp. 4196–4191, 2003.

[16] J. A. Carretero and M. A. Nahon, "Solving minimum distance problems with convex or concave bodies using combinatorial global optimization algorithms," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 35, no. 6, pp. 1144–1155, 2005.

[17] H. G. Zhang, Z. W. Liu, G.-B. Huang, and Z. S. Wang, "Novel weighting-delay-based stability criteria for recurrent neural networks with time-varying delay," *IEEE Transactions on Neural Networks*, vol. 21, no. 1, pp. 91–106, 2010.

[18] Y. M. Zhong, S. Bijan, and Y. Tian, "A new neural network for robot path planning," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM '08)*, pp. 1361–1366, August 2008.

[19] H. X. Zhang, J. González-Gómez, S. Y. Chen et al., "A novel modular climbing caterpillar using low-frequency vibrating passive suckers," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 412–523, September 2007.

[20] W.-B. Young and W.-H. Wu, "Optimization of the skin thickness distribution in the composite wind turbine blade," in *Proceedings of the International Conference on Fluid Power and Mechatronics (FPM '11)*, pp. 62–66, August 2011.

[21] H. H. Szu, "Non-convex optimization," in *Proceedings of the SPIE Real-Time Signal*, vol. 0698, pp. 59–65, San Diego, Calif, USA, Aughust, 1986.

[22] H. Szu and R. Hartley, "Fast simulated annealing," *Physics Letters A*, vol. 122, no. 3-4, pp. 157–162, 1987.

[23] L. Ingber, "Very fast simulated re-annealing," *Mathematical and Computer Modelling*, vol. 12, no. 8, pp. 967–973, 1989.

[24] L. Ingber, "Simulated annealing: practice versus theory," *Mathematical and Computer Modelling*, vol. 18, no. 11, pp. 29–57, 1993.

[25] Y. Ruoli and G. Jifa, "An efficient algorithm for simulated annealing global optimizing," *System Engineering Theory and Practice*, vol. 5, no. 8, pp. 29–35, 1997.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 286074, 14 pages
http://dx.doi.org/10.1155/2013/286074

Research Article

Dynamic Modeling and Fuzzy Self-Tuning Disturbance Decoupling Control for a 3-DOF Serial-Parallel Hybrid Humanoid Arm

Yueling Wang, Runjie Shi, and Hongbin Wang

School of Control Science and Engineering, Yanshan University, Qinhuangdao 066004, China

Correspondence should be addressed to Yueling Wang; yuelingw@ysu.edu.cn

Received 25 June 2013; Accepted 3 October 2013

Academic Editor: Amir H. Alavi

A unique fuzzy self-tuning disturbance decoupling controller (FSDDC) is designed for a serial-parallel hybrid humanoid arm (HHA) to implement the throwing trajectory-tracking mission. Firstly, the dynamic model of the HHA is established and the input signal of the throwing process is obtained by studying the throwing process of human's arm. Secondly, the FSDDC, incorporating the disturbance decoupling controller (DDC) and the fuzzy logic controller (FLC), is designed to ensure trajectory tracking of the HHA in the presence of uncertainties and disturbances. With the FSDDC method, the HHA system can be decoupled by actively estimating and rejecting the effects of both the internal plant dynamics and external disturbances. The self-tuning parameters are adapted online to improve the performance of the FSDDC; thus, it does not require detailed system parameters of the presented FSDDC. Finally, the controller introduced is compared with a PD controller which is commonly used for the robot manipulators control in industry. The effectiveness of the designed FSDDC is illustrated by simulations.

1. Introduction

Because of the great advantages of serial robots and parallel robots where the former have simple structures, big workspaces, and easy control, the latter are in high force-to-weight, high stiffness and accuracy, and low inertia; they are widely used in various fields such as industrial production and scientific experiments [1–4]. Since robot is a very complicated multiple-input multiple-output (MIMO) nonlinear system with time-varying, strong-coupling characteristics, the design of robust controllers which is suitable for real-time control of manipulators is one of the most challenging tasks for many control engineers, especially when manipulators are required to maneuver very quickly within external disturbances. Various advanced control strategies, either model-based control or model-free control, have been applied in a variety of robots to improve the control performance [5–13].

The 3-DOF serial-parallel hybrid humanoid arm (HHA) combines the characteristics of series and parallel robot, as shown in Figure 1, and can be used in humanoid robots or automated production lines due to its special serial-parallel hybrid structure and performance, compared with other parallel or serial robots. However, the special structure of the HHA introduces complexity in kinematics, dynamic equations, and coupling of the system, placing greater demands on control methods. Proportional and derivative (PD) controller [14, 15], for its simple and effective properties, does not require any knowledge of the robot dynamics and is widely used in industrial robots, but it cannot follow the desire of accuracy and robustness. The combination of modern control methods and intelligent control technique [5–8] is widely used in various robot control research and brings out better effect. However, model-based controllers [9] cannot be accurately applied since it is almost impossible to obtain exact dynamic models. Although adaptive control [7, 8, 10] can achieve satisfied control effects and compensate for partially unknown manipulator dynamics and sliding-mode control [6–10] is robust with respect to system uncertainties and external disturbance, they all need prior knowledge of the robot. A robust adaptive trajectory tracking controller based on neural network algorithm is designed for HHA in [16], but

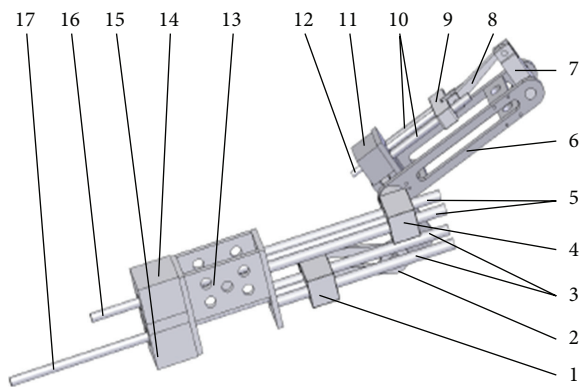


Figure 1: The prototype of 3-DOF forearm.

the coupling effects are not taken into account in these control schemes. The coupling effects can limit the performances of HHA, since the dynamic coupling may restrict not only the further improvement of control performances in HHA but also the development of potential of such robot [11–13]. Eliminating or reducing the coupling of 3-DOF serial-parallel hybrid humanoid arm can improve the system control accuracy and the trajectory tracking performance. From the above analysis, it can be concluded that a control approach, suitable for multivariable systems with great disturbance rejection, decoupling ability, and model-free and strong robustness, is urgently needed for the HHA.

In recent years, a dynamic disturbance decoupling control (DDC) [17] approach can be applied to the manipulators. This disturbance decoupling control method is rooted in a new ground-breaking paradigm of control design: active disturbance rejection control (ADRC). The original concept of active disturbance rejection was proposed for the nonlinear structure by Han [18–20] and was further simplified and parameterized by Gao [21, 22]. The active disturbance rejection control (ADRC) method is a new kind of practical digital control technology which does not depend upon the accurate mathematical model of the plant. The central idea is that both the internal dynamics and external disturbances of the system can be estimated by extended state observe (ESO) [21, 23, 24] and compensated for in real time. Then the system can be reduced to the standard form of linear system, a simple unit gain of integrals, making the control problem much easier. More importantly, with the proposed parameterization of active disturbance rejection control (ADRC), it becomes a viable candidate for robot decoupling control [17] and practical applications [25–28].

With the continuous development of the fuzzy logic control theory, its application on robots received more and more attentions [5–7]. The main advantage of FLC comes from the applicability to systems when the mathematical model is not known exactly, and another important advance of fuzzy controller is the short rise time and small overshoot [29–31]. The knowledge about system characteristics coming from experts can be expressed by fuzzy logic rules, which makes this control method more attractive than the others.

To enhance the controller performance, in this study, hybridization of these two controller structures comes to one mind immediately to exploit the beneficial sides of both categories, known as a fuzzy self-tuning disturbance decoupling controller (FSDDC). The parameters of the normalization factor of output variable in the fuzzy mechanism and the control gain tuned by fuzzy logic units are adapted online to ensure the performance of the fuzzy self-tuning disturbance decoupling control system. This control algorithm can be applied to manipulator systems with unmodeled dynamics, unstructured uncertainties, decoupling, and external disturbance.

The remainder of the paper is organized as follows. In Section 2, the structure and the throwing process of 3-DOF HHA are analyzed by the means of numerical mathematics and image processing. Mathematical modeling is presented in Section 3. In Section 4, the DDC and FSDDC for HHA are proposed. Computer simulation results of the proposed FSDDC, DDC, and PD algorithms of the HHA are given in Section 5. Finally, Section 6 concludes the paper.

2. 3-DOF Serial-Parallel Hybrid Humanoid Arm

2.1. The Structure of 3-DOF HHA. This 3-DOF serial-parallel hybrid humanoid arm is composed of a 2-DOF parallel mechanism which forms the upper arm of HHA and a 1-DOF serial mechanism which forms the forearm of HHA.

As shown in Figure 1, the 2-DOF parallel mechanism is composed of linear motors 14 and 15 and their movers 16 and 17, upper arm orbits 3 and 5, upper arm slip tubes 1 and 4, elbow handspikes 2, and upper arm pedestal 13. The 1-DOF serial mechanism is composed of linear motor 11 and its mover 12, forearm pedestal 11, wrist handspike 8, wrist 7, forearm orbit 10, and forearm slip tube 9. The motors 14 and 15 and upper arm orbit are fixed on upper arm pedestal. The motor 11 is fixed on forearm pedestal. The movers P1, P2, and P3 move forwards or backwards relatively to its main-body motors M1, M2, and M3, and the movement of motors drives the other components to achieve the desired action through the rotation axes.

The coordinates $\{O_0\}$ and $\{O_1\}$ are shown in Figure 2. $\{O_0\}$ is the fixed reference coordinate system, and $\{O_1\}$ is attached to the upper arm of HHA. Referenced to the horizontal, the upward inclination of the upper arm is up about 20°.

The same point $\mathbf{p}$ is described in $\{O_0\}$ and $\{O_1\}$: ${}^0\mathbf{p}$ and ${}^1\mathbf{p}$ have transformation relations as follows:

$$ {}^0\mathbf{p} = \mathbf{R}\,{}^1\mathbf{p}, \tag{1} $$

where

$$ \mathbf{R} = \begin{bmatrix} \cos 20^\circ & -\sin 20^\circ \\ \sin 20^\circ & \cos 20^\circ \end{bmatrix} = \begin{bmatrix} 0.940 & -0.342 \\ 0.342 & 0.940 \end{bmatrix}. \tag{2} $$

Under the assumption of the regular shape, smooth surface, and linear density steady of every components, the sketch of HHA in $\{O_0\}$ and $\{O_1\}$ is shown in Figure 2. The dot Q, O, G is the center of mass of the motors M1, M2, and

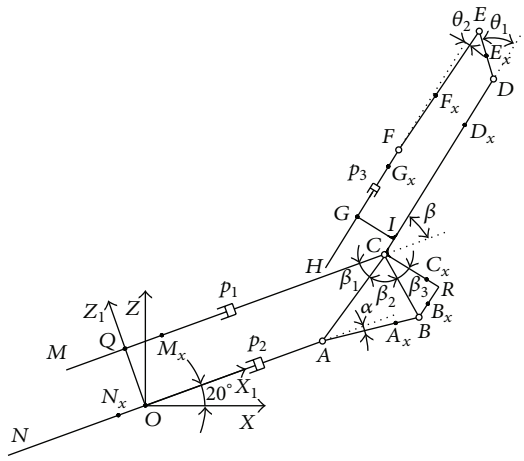


Figure 2: Structure of forearm in $\{O_0\}$ and $\{O_1\}$.

M3 and their pedestals, respectively. The dot C, A, F is the center of mass of the slip tubes S1, S2, and S3, respectively. The lengths of AN and CM are both l_1, and the lengths of OQ, AB, RC, RB, CD, BC, EF, DE, ID, and FH are l_0, l_2, l_3, l_4, l_5, l_6, l_{11}, l_{10}, l_{15}, and l_{30}, respectively. The lengths of QC, OA, and GF are defined as x_1, x_2, and x_3, respectively.

In coordinates $\{O_1\}$ and $\{O_0\}$, the position vectors of dot A and C are

$$\begin{aligned} {}^{O_1}\mathbf{p}_A &= [x_2 \quad 0]^T, & {}^{O_1}\mathbf{p}_C &= [x_1 \quad l_1]^T, \\ {}^{O_0}\mathbf{p}_A &= \mathbf{R}[x_2 \quad 0]^T, & {}^{O_0}\mathbf{p}_C &= \mathbf{R}[x_1 \quad l_1]^T. \end{aligned} \tag{3}$$

As shown in Figure 2, β is the angle between CD and the axis X_1. α is the angle between AB and axis X_1. The HHA is on the initial state when $\beta = 0$.

Define

$$\begin{gathered} \Delta x = x_1 - x_2, \qquad l_7 = l_{15} - x_3, \qquad l_8 = \sqrt{l_1^2 + \Delta^2 x}, \\ l_9 = \sqrt{4l_6^2 l_8^2 - \left(l_8^2 + l_6^2 - l_2^2\right)^2}, \qquad l_{16} = \sqrt{l_{10}^2 + l_7^2}, \\ l_{20} = \sqrt{l_{10}^2 + l_{16}^2 - l_{11}^2}, \qquad l_{22} = \sqrt{l_{11}^2 + l_{16}^2 - l_{10}^2}, \\ \mathbf{\Omega} = [\cos\alpha \quad -\sin\alpha]^T, \qquad \overline{\mathbf{\Omega}} = [\sin\alpha \quad \cos\alpha]^T, \\ \mathbf{\Theta}_1 = [-\sin\beta \quad \cos\beta]^T, \qquad \overline{\mathbf{\Theta}} = [\cos\beta \quad \sin\beta]^T, \\ \mathbf{\Phi}_1 = [\cos\theta_1 \quad \sin\theta_1]^T, \qquad \mathbf{\Phi}_2 = [\cos\theta_2 \quad -\sin\theta_2]^T, \\ \mathbf{e}_1 = [1 \quad 0]^T, \qquad \mathbf{e}_2 = [0 \quad 1]^T, \\ \mathbf{\Theta} = \left[\overline{\mathbf{\Theta}} \quad \mathbf{\Theta}_1\right], \qquad \mathbf{L}_{34} = [l_3 \quad l_4], \qquad \mathbf{E} = [\mathbf{e}_1 \quad \mathbf{e}_2]. \end{gathered} \tag{4}$$

As shown in Figure 2, the angles β and α can be described as follows:

$$\begin{gathered} \beta = \beta_1 + \beta_2 + \beta_3 - \frac{\pi}{2}, \\ \beta_1 = \operatorname{arctg}\left(\frac{l_1}{\Delta x}\right), \qquad \beta_3 = \operatorname{arctg}\left(\frac{l_4}{l_1}\right), \\ \beta_2 = \arccos\left(\frac{l_8^2 + l_6^2 - l_2^2}{2l_6 l_8}\right), \\ \alpha = \operatorname{arctg}\frac{l_1 - \mathbf{L}_{34}\overline{\mathbf{\Theta}}}{\Delta x - \mathbf{L}_{34}\mathbf{\Theta}_1}. \end{gathered} \tag{5}$$

The angle θ_1 which is the angle between DE and CD and the angle θ_2 which is the angle between EF and GF can be described as follows:

$$\begin{gathered} \theta_1 = \arccos\left(-\frac{l_7}{l_{16}}\right) - \arccos\left(\frac{l_{20}^2}{(2l_{10}l_{16})}\right), \\ \theta_2 = \arccos\left(\frac{l_7}{l_{16}}\right) - \arccos\left(\frac{l_{22}^2}{(2l_{11}l_{16})}\right). \end{gathered} \tag{6}$$

In coordinates $\{O_0\}$, the position vectors of D, R, G, and F can be described as follows:

$$\begin{gathered} {}^{O_0}\mathbf{p}_D = \mathbf{R}\left({}^{O_1}\mathbf{p}_C + l_5\overline{\mathbf{\Theta}}\right), \qquad {}^{O_0}\mathbf{p}_R = \mathbf{R}\left({}^{O_1}\mathbf{p}_C - l_3\mathbf{\Theta}_1\right), \\ {}^{O_0}\mathbf{p}_G = \mathbf{R}\left({}^{O_1}\mathbf{p}_C + (l_5 - l_{15})\overline{\mathbf{\Theta}} + l_{10}\mathbf{\Theta}_1\right), \\ {}^{O_0}\mathbf{p}_F = \mathbf{R}\left({}^{O_1}\mathbf{p}_G + x_3\overline{\mathbf{\Theta}}\right). \end{gathered} \tag{7}$$

Ax, Bx, Cx, and Dx are arbitrary dots of AB, RB, RC, and CD. In coordinates $\{O_0\}$, their position vectors are

$$\begin{aligned} {}^{O_0}\mathbf{p}_{Ax} &= \mathbf{R}\left({}^{O_1}\mathbf{p}_A + x\mathbf{\Omega}\right), & 0 \le x \le l_2, \\ {}^{O_0}\mathbf{p}_{Bx} &= \mathbf{R}\left({}^{O_1}\mathbf{p}_R - x\overline{\mathbf{\Theta}}\right), & 0 \le x \le l_4, \\ {}^{O_0}\mathbf{p}_{Cx} &= \mathbf{R}\left({}^{O_1}\mathbf{p}_C - x\mathbf{\Theta}_1\right), & 0 \le x \le l_3, \\ {}^{O_0}\mathbf{p}_{Dx} &= \mathbf{R}\left({}^{O_1}\mathbf{p}_C + x\overline{\mathbf{\Theta}}\right), & 0 \le x \le l_5. \end{aligned} \tag{8}$$

Mx and Nx are arbitrary dots in CN and AN. In coordinates $\{O_0\}$, their position vectors are

$$\begin{aligned} {}^{O_0}\mathbf{p}_{Mx} &= \mathbf{R}\left({}^{O_1}\mathbf{p}_A - x\mathbf{e}_1\right), & 0 < x < l_1, \\ {}^{O_0}\mathbf{p}_{Nx} &= \mathbf{R}\left({}^{O_1}\mathbf{p}_C + x\mathbf{e}_1\right), & 0 \le x \le l_1. \end{aligned} \tag{9}$$

Gx, Fx, and Ex are arbitrary dots in HF, EF, and DE. In coordinates $\{O_0\}$, their position vectors are

$$\begin{aligned} {}^{O_0}\mathbf{p}_{Gx} &= \mathbf{R}\left({}^{O_1}\mathbf{p}_F - x\overline{\mathbf{\Theta}}\right), & 0 < x < l_{30}, \\ {}^{O_0}\mathbf{p}_{Fx} &= \mathbf{R}\left({}^{O_1}\mathbf{p}_F + x\mathbf{\Theta}\mathbf{\Phi}_2\right), & 0 \le x \le l_{11}, \\ {}^{O_0}\mathbf{p}_{Ex} &= \mathbf{R}\left({}^{O_1}\mathbf{p}_D + x\mathbf{\Theta}\mathbf{\Phi}_1\right), & 0 \le x \le l_{10}. \end{aligned} \tag{10}$$

Table 1: Elbow joint variables data in raising stage.

Variables	Values											Unit
β	0	9	17	27	35	46	55	62	72	82	90	°
Δx	0.2	0.185	0.171	0.151	0.139	0.120	0.103	0.095	0.079	0.066	0.055	m
x_1	0.3	0.292	0.285	0.275	0.269	0.26	0.251	0.247	0.239	0.233	0.227	m
x_2	0.1	0.107	0.114	0.124	0.130	0.14	0.148	0.152	0.160	0.167	0.172	m
t	0	0.12	0.24	0.36	0.48	0.6	0.72	0.84	0.96	1.08	1.2	s

Table 2: Elbow joint variables data in throwing stage.

Variables	Values											Unit
β	90	70	57	43	33	24	15	9	5	2	0	°
Δx	0.055	0.082	0.102	0.125	0.142	0.159	0.174	0.185	0.192	0.197	0.2	m
x_1	0.227	0.241	0.251	0.262	0.271	0.279	0.287	0.292	0.296	0.298	0.3	m
x_2	0.172	0.159	0.149	0.137	0.129	0.120	0.113	0.107	0.104	0.101	0.1	m
t	1.5	1.58	1.66	1.74	1.822	1.9	1.98	2.06	2.14	2.22	2.3	s

2.2. Analysis and Reference Trajectory of Throwing Process. A complete throwing process of HHA in a cycle consists of three stages: HHA raising from initial state stage, HHA waiting for the load stage, and HHA throwing the load out stage. Whether the object can be thrown at a certain position and a certain speed is a key part in the complete throwing process, and obtaining the corresponding reference signal is considered as a difficult problem.

To obtain the corresponding reference signal, a method we proposed is shown as follows. Firstly, the changing curve of HHA elbow joint angle, wrist joint angle, and time should be found by analyzing the throwing process of man's arm, respectively, and then the changing curve of Δx, x_3 and time can be found through formula (5) and (6), respectively. By planning the motion of S1 and S2, we can get the changing curve of x_1, x_2, x_3, and time, respectively.

To study the man's throwing process in detail, a camera is used to get the video image of raising and throwing process. After processing by MATLAB and Photoshop, the video image can be divided into four parts: the raising of forearm, the putting down of forearm, the raising of hand, and the putting down of hand. Define the beginning time of raising process at 0 seconds and the beginning time of throwing process at 1.5 seconds; the video image processing results are shown in Figure 3.

Considering the structure of the elbow joint and energy saving, the Δx can be produced by the following ways: when the forearm is raising, S1 and S2 move close to each other; when the forearm is putting down, S1 and S2 move away from each other. Then, the relationship between variables β, Δx, x_1, x_2, and time are shown in Tables 1 and 2; the relationship between variables θ_1, x_3, and time are shown in Tables 3 and 4, respectively. Processing the data in Tables 1–4 by MATLAB CFtool, we get the changing curve of x_1, x_2, x_3 time in a cycle as shown in Figure 4 and their fragment expressions are given as follows:

$$x_1 = \begin{cases} 0.01t^2 - 0.073t + 0.3, & 0 \le t < 1.2\text{s} \\ 0.227, & 1.2\text{s} \le t \le 1.5\text{s} \\ -0.099t^2 + 0.468t - 0.252, & 1.5\text{s} \le t < 2.3\text{s}, \end{cases}$$

$$x_2 = \begin{cases} -0.009t^2 + 0.071t + 0.1, & 0 \le t < 1.2\text{s} \\ 0.172, & 1.2\text{s} \le t \le 1.5\text{s} \\ 0.099t^2 - 0.468t + 0.649, & 1.5\text{s} \le t \le 2.3\text{s}, \end{cases}$$

$$x_3 = \begin{cases} -0.027t^2 - 0.032t + 0.166, & 0 \le t < 1.08\text{s} \\ 0.1, & 1.08\text{s} \le t \le 1.5\text{s} \\ 0.2e^{-0.358(x-1.5)} - 0.1e^{-6.45(x-1.5)}, & 1.5\text{s} \le t < 1.86\text{s} \\ 0.166, & 1.86\text{s} \le t \le 2.3\text{s}. \end{cases} \tag{11}$$

3. Dynamic Modeling for HHA

In this section, the dynamic model of the HHA is derived. The Lagrange formalism used for the serial-parallel mechanism can be written as follows:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial L}{\partial \mathbf{q}} = \boldsymbol{\tau}, \tag{12}$$

where $L = V - P$ is the Lagrange function, K and P are the kinetic and potential energy functions, respectively, $\mathbf{q} = [q_1 \ q_2 \ q_3]^T$, $\dot{\mathbf{q}} = [\dot{q}_1 \ \dot{q}_2 \ \dot{q}_3]^T$ are the vectors of the slip tubes position and velocity $\boldsymbol{\tau}^{3\times 3}$ is the vector of the corresponding control input forces.

3.1. The Computing for Kinetic and Potential Energy of Every Component. The total kinetic and potential energy of movers P1, P2, and P3 is given, respectively, as

$$\begin{aligned} V_1 &= \frac{1}{2}\rho_1 \int_0^{l_1} \left({}^{O_0}\dot{\mathbf{p}}_{Mx}^T \, {}^{O_0}\dot{\mathbf{p}}_{Mx} + {}^{O_0}\dot{\mathbf{p}}_{Nx}^T \, {}^{O_0}\dot{\mathbf{p}}_{Nx} \right) \mathrm{d}x, \\ P_1 &= \rho_1 g \int_0^{l_1} \left(\mathbf{e}_2^T \, {}^{O_0}\mathbf{p}_{Mx} + \mathbf{e}_2^T \, {}^{O_0}\mathbf{p}_{Nx} \right) \mathrm{d}x, \\ V_2 &= \frac{1}{2}\rho_1 \int_0^{l_{30}} {}^{O_0}\dot{\mathbf{p}}_{Gx}^T \, {}^{O_0}\dot{\mathbf{p}}_{Gx} \, \mathrm{d}x, \\ P_2 &= \rho_1 g \int_0^{l_{30}} \mathbf{e}_2^T \, {}^{O_0}\mathbf{p}_{Gx} \, \mathrm{d}x, \end{aligned} \tag{13}$$

where ρ_1 is the linear density of P1, P2, and P3.

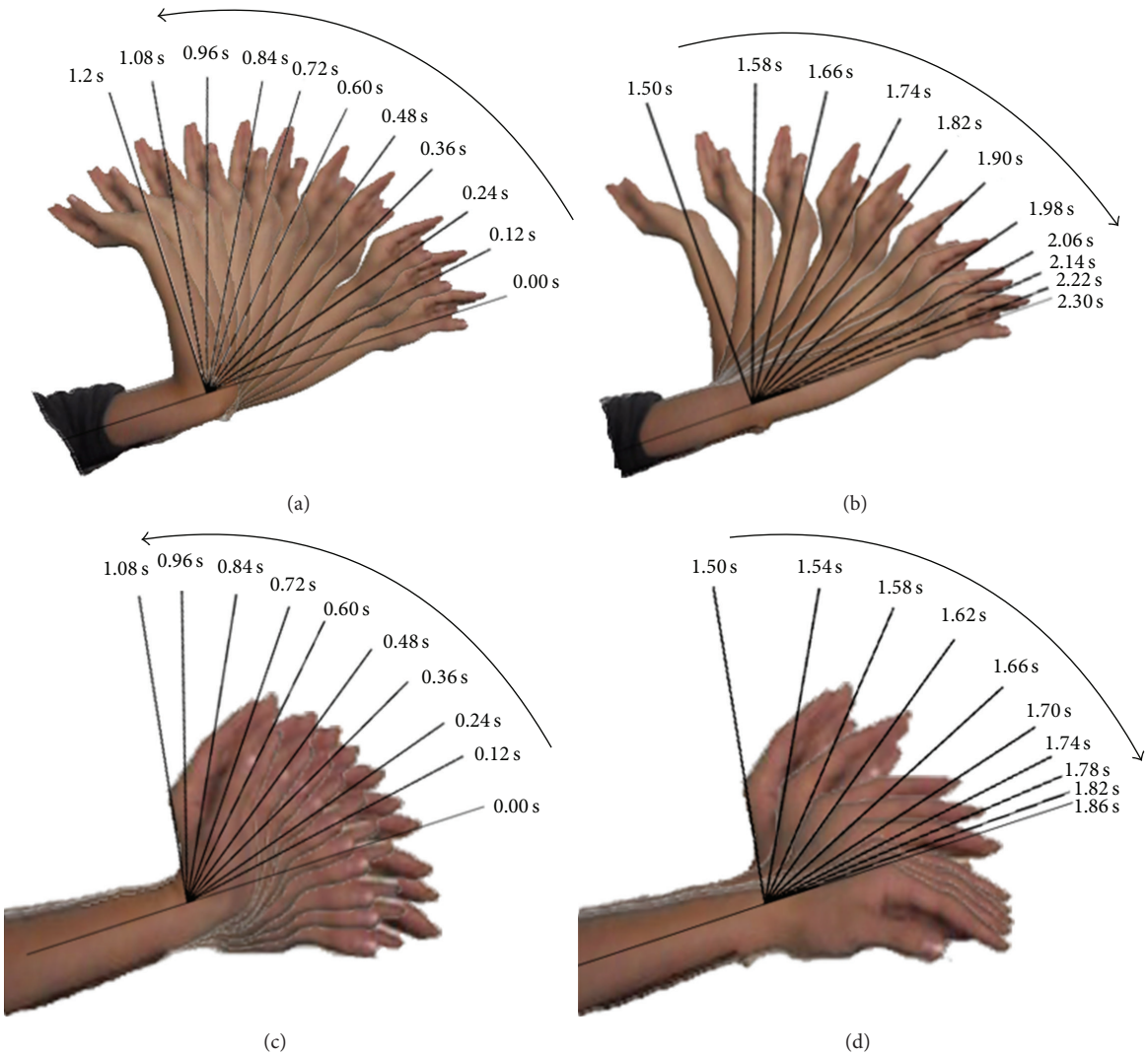


Figure 3: Video image processing results of complete throwing process. (a) Forearm raising, (b) forearm putting down, (c) hand raising, and (d) hand putting down video image processing result.

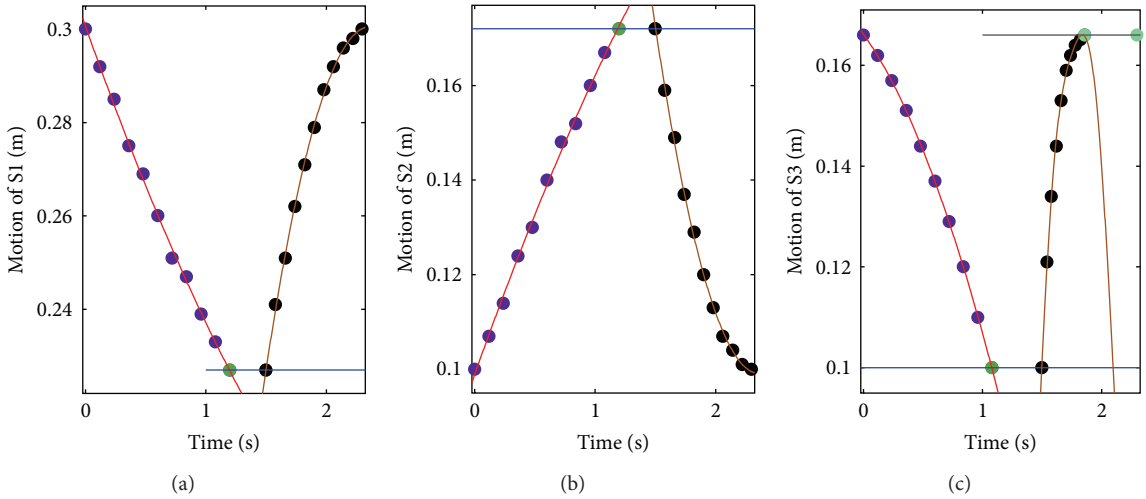


Figure 4: Position motion of the slip tubes. (a) The motion of S1, (b) the motion of S2, and (c) the motion of S3.

Table 3: Wrist joint variables data in raising stage.

Variables	Values										Unit
θ_1	0	10	17	27	36	46	53	62	73	81	°
x_3	0.166	0.162	0.157	0.151	0.144	0.137	0.129	0.120	0.11	0.1	m
t	0	0.12	0.24	0.36	0.48	0.6	0.72	0.84	0.96	1.08	s

Table 4: Wrist joint variables data in throwing stage.

Variables	Values										Unit
θ_1	81	62	48	36	24	15	9	5	2	0	°
x_3	0.1	0.121	0.134	0.144	0.153	0.159	0.162	0.164	0.165	0.166	m
t	1.5	1.54	1.58	1.62	1.66	1.7	1.74	1.78	1.82	1.86	s

Table 5: Fuzzy rules for Δk_{pi}.

e_i	$\dot{e}_i$						
	NB	NM	NS	ZO	PS	PM	PB
NB	PB	PB	PM	PB	NM	ZO	ZO
NM	PB	PB	PM	PS	PS	ZO	NS
NS	PM	PM	PS	PS	NS	NS	NS
ZO	PM	PM	NS	PB	NS	NM	NM
PS	PS	PS	NS	NS	NS	NM	NM
PM	PS	NM	NS	NM	NM	NM	NB
PB	ZO	ZO	NM	PB	NM	NB	NB

Table 6: Fuzzy rules for Δk_{di}.

e_i	$\dot{e}_i$						
	NB	NM	NS	ZO	PS	PM	PB
NB	PS	NS	NB	NB	PM	ZO	PS
NM	PS	NS	NB	NM	NM	ZO	ZO
NS	ZO	NS	NM	NM	PS	NS	ZO
ZO	ZO	NS	NS	PB	NS	NS	ZO
PS	ZO	ZO	PS	ZO	ZO	ZO	ZO
PM	PB	NS	PS	PS	PS	PS	PB
PB	PB	PM	PM	NB	PS	PS	PB

The kinetic and potential energy of part CD and wrist handspike is given, respectively, as

$$\begin{aligned} V_3 &= \frac{1}{2}\rho_2 \int_0^{l_5} {}^{O_0}\dot{\mathbf{p}}_{Dx}^T \, {}^{O_0}\dot{\mathbf{p}}_{Dx} \, \mathrm{d}x, \\ P_3 &= \rho_2 g \int_0^{l_5} \mathbf{e}_2^T \, {}^{O_0}\mathbf{p}_{Dx} \, \mathrm{d}x, \\ V_4 &= \frac{1}{2}\rho_3 \int_0^{l_{10}} {}^{O_0}\dot{\mathbf{p}}_{Ex}^T \, {}^{O_0}\dot{\mathbf{p}}_{Ex} \, \mathrm{d}x, \\ P_4 &= \rho_3 g \int_0^{l_{10}} \mathbf{e}_2^T \, {}^{O_0}\mathbf{p}_{Ex} \, \mathrm{d}x, \end{aligned} \tag{14}$$

where ρ_2 and ρ_3 are the linear density of CD and wrist handspike, respectively.

Table 7: Parameters of the HHA.

Parameters	Values	Unit
l_0	0.5	m
l_1	0.09	m
l_2	0.15	m
l_3	0.09	m
l_4	0.05	m
l_5	0.3	m
l_6	0.103	m
l_{10}	0.065	m
l_{11}	0.19	m
l_{15}	0.28	m
l_{30}	0.19	m
ρ_1	1	kg/m
ρ_2	1.5	kg/m
ρ_3	2	kg/m
ρ_4	1	kg/m
m_9	0.5	kg
g	9.8	m/s^2

Table 8: Parameters of the FSDDC.

Gains	Values	Gains	Values	Gains	Values
b_{10}	1	b_{20}	1.3	b_{30}	1.5
w_{01}	80	w_{02}	80	w_{03}	60
K_{p1}	2200	K_{p2}	2500	K_{p3}	1600
K_{d1}	97	K_{d2}	100	K_{d3}	60
ζ_1	400	ζ_2	500	ζ_3	300
η_1	9	η_2	10	η_3	6

The kinetic and potential energy of parts AB, BR, RC, and EF of forearm pedestal is given, respectively, as

$$\begin{aligned} V_5 &= \frac{1}{2}\rho_4 \int_0^{l_2} {}^{O_0}\dot{\mathbf{p}}_{Ax}^T \, {}^{O_0}\dot{\mathbf{p}}_{Ax} \, \mathrm{d}x, \\ P_5 &= \rho_4 g \int_0^{l_2} \mathbf{e}_2^T \, {}^{O_0}\mathbf{p}_{Ax} \, \mathrm{d}x, \\ V_6 &= \frac{1}{2}\rho_4 \int_0^{l_4} {}^{O_0}\dot{\mathbf{p}}_{Bx}^T \, {}^{O_0}\dot{\mathbf{p}}_{Bx} \, \mathrm{d}x, \end{aligned}$$

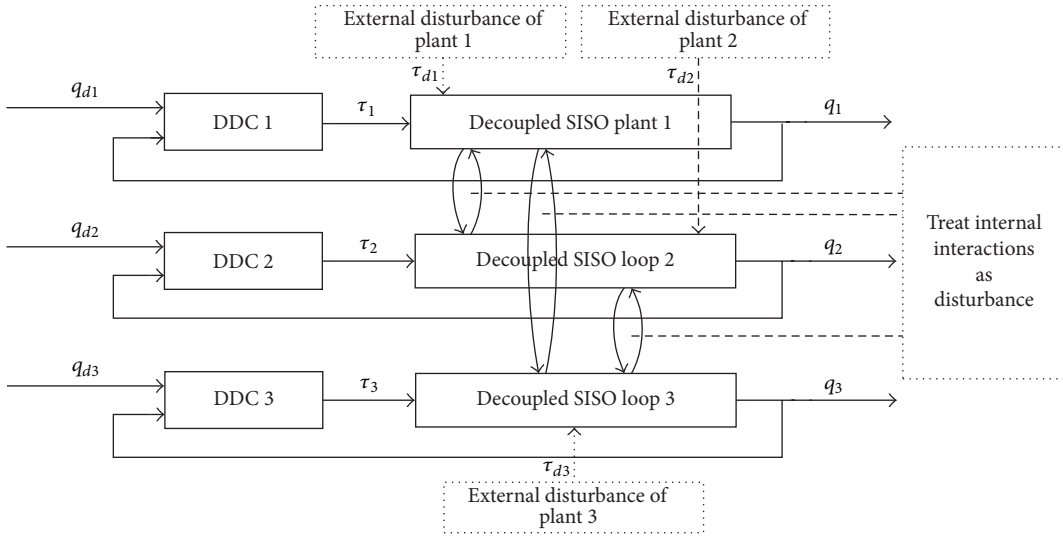


Figure 5: Diagram of the DDC for HHA.

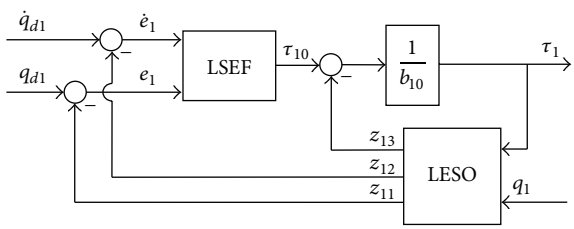


Figure 6: The structure of DDC for first loop.

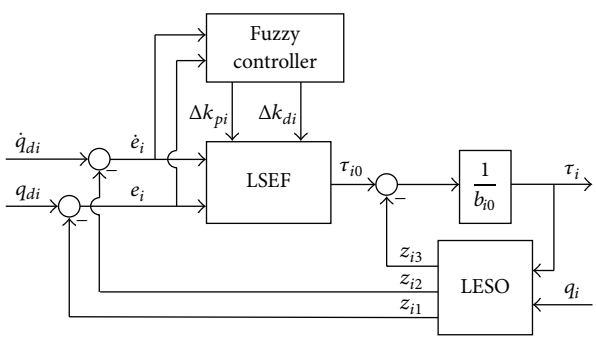


Figure 7: The structure of FSDDC.

$$P_6 = \rho_4 g \int_0^{l_4} \mathbf{e}_2^T \, {}^{O_0}\mathbf{p}_{Bx} \, \mathrm{d}x,$$

$$V_7 = \frac{1}{2}\rho_4 \int_0^{l_3} {}^{O_0}\dot{\mathbf{p}}_{Cx}^T \, {}^{O_0}\dot{\mathbf{p}}_{Cx} \, \mathrm{d}x,$$

$$P_7 = \rho_4 g \int_0^{l_3} \mathbf{e}_2^T \, {}^{O_0}\mathbf{p}_{Cx} \, \mathrm{d}x, \tag{15}$$

$$V_8 = \frac{1}{2}\rho_4 \int_0^{l_{11}} {}^{O_0}\dot{\mathbf{p}}_{Fx}^T \, {}^{O_0}\dot{\mathbf{p}}_{Fx} \, \mathrm{d}x,$$

$$P_8 = \rho_4 g \int_0^{l_{11}} \mathbf{e}_2^T \, {}^{O_0}\mathbf{p}_{Fx} \, \mathrm{d}x,$$

where ρ_2 is the linear density of AB, BR, RC, and EF.

Table 9: Parameters of the DDC.

Gains	Values	Gains	Values	Gains	Values
b_{10}	1	b_{20}	1.3	b_{30}	1.5
w_{01}	80	w_{02}	80	w_{03}	60
K_{p1}	2200	K_{p2}	2500	K_{p3}	1600
K_{d1}	97	K_{d2}	100	K_{d3}	60

Table 10: Parameters of the PD controller.

Gains	Values	Gains	Values	Gains	Values
K_{p1}	2200	K_{p2}	1800	K_{p3}	1300
K_{d1}	95	K_{d2}	85	K_{d3}	25

The total kinetic and potential energy of motor M3 and its pedestal is given as

$$V_9 = \frac{1}{2} m_6 \, {}^{O_0}\dot{\mathbf{p}}_Q^T \, {}^{O_0}\dot{\mathbf{p}}_Q, \qquad P_9 = m_6 g \mathbf{e}_2^T \, {}^{O_0}\mathbf{p}_Q, \tag{16}$$

where m_9 is the mass of M3 and its pedestal.

3.2. The Dynamic Model of the HHA. The dynamics of the HHA can be written as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \boldsymbol{\tau}_d = \boldsymbol{\tau}, \tag{17}$$

where $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}} \in R^3$ are the vectors of slip tubes position, velocity, and acceleration, respectively, $\mathbf{M}(\mathbf{q}) = \mathbf{M}_0(\mathbf{q}) + \Delta\mathbf{M}(\mathbf{q}) \in R^{3\times 3}$ is a symmetric positive definite inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}}) + \Delta\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in R^{3\times 3}$ is the matrix of centripetal and coriolis force, $\mathbf{G}(\mathbf{q}) = \mathbf{G}_0(\mathbf{q}) + \Delta\mathbf{G}(\mathbf{q}) \in R^3$ is the vector of gravitational force, $\boldsymbol{\tau}_d \in R^3$ is the vector of generalized input due to external disturbance with $\|\boldsymbol{\tau}_d\| \le d$, and $\boldsymbol{\tau} \in R^3$ is the input force vector. Here $\mathbf{M}_0(\mathbf{q})$, $\mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}})$, and $\mathbf{G}_0(\mathbf{q})$ are the nominal parts calculated by Lagrange method, where $\Delta\mathbf{M}_0(\mathbf{q})$, $\Delta\mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}})$, and $\Delta\mathbf{G}_0(\mathbf{q})$

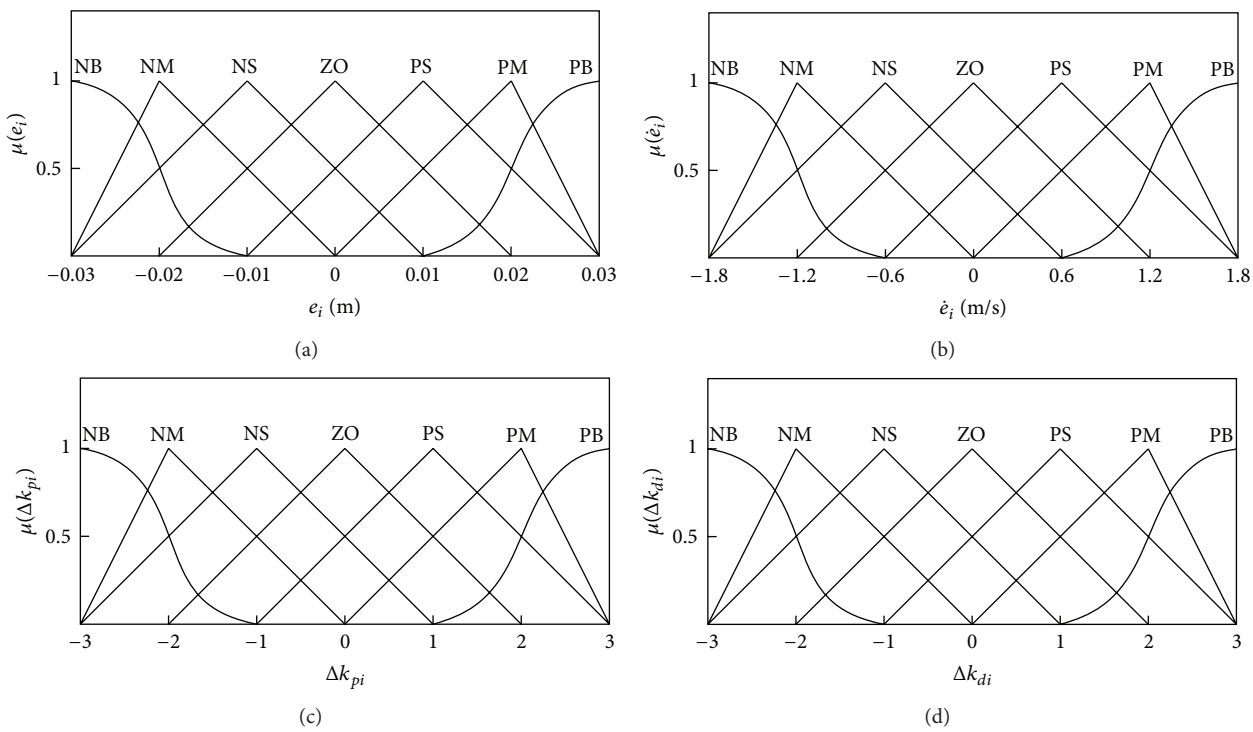


Figure 8: Membership functions: (a) error, (b) derivative of error, (c) gains of Δk_{pi}, and (d) gains of Δk_{di}.

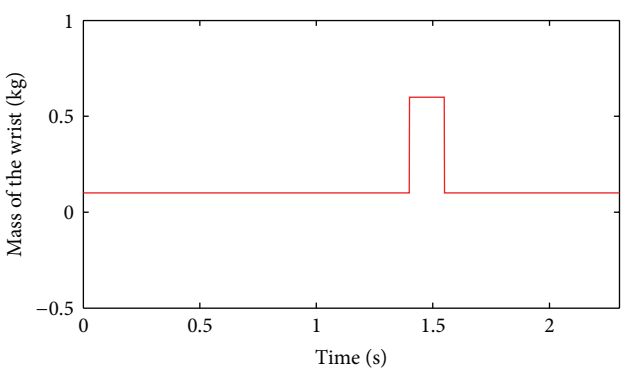


Figure 9: Variation of the mass of the wrist.

represent the perturbations in the system matrices. Then the dynamic model of HHA can be rewritten as

$$\mathbf{M}_0(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_0(\mathbf{q}) + \boldsymbol{\tau}_d + \mathbf{F}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}) = \boldsymbol{\tau}, \quad (18)$$

where

$$\mathbf{F}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}}) = \Delta\mathbf{M}_0(\mathbf{q})\ddot{\mathbf{q}} + \Delta\mathbf{C}_0(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \Delta\mathbf{G}_0(\mathbf{q}) \in R^3 \quad (19)$$

is the lumped system uncertainty and satisfies [9]

$$\|\mathbf{F}(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}})\| \le b_0 + b_1\|\mathbf{q}\| + b_2\|\dot{\mathbf{q}}\|^2. \quad (20)$$

4. Fuzzy Self-Tuning Disturbance Decoupling Control (FSDDC) for HHA

In this section, the FSDDC for HHA is developed, which combines the advantages of the DDC and the FLC. First, the DDC for HHA is derived. Then, the FSDDC is proposed to establish the FLC for the proposed DDC.

4.1. Disturbance Decoupling Control (DDC) for HHA. DDC is a relatively new control design concept and a natural fit for the purpose of disturbance decoupling in robot systems. As it is applied to the control of the HHA system, the idea is that all the nonlinear, time-varying uncertainties and coupling terms are parts of the total disturbance and can be actively estimated using the extended state observer (ESO) and canceled in the control law, leading to roughly three single-input and single-output plants. In this section, three disturbance decoupling controllers are designed for HHA to do trajectory tracking.

The realistic model (18) can be reformulated as

$$\ddot{\mathbf{q}} = -\mathbf{M}_0^{-1}\left(\mathbf{C}_0\dot{\mathbf{q}} + \mathbf{G}_0 + \boldsymbol{\tau}_d + \mathbf{F}\right) + \mathbf{M}_0^{-1}\boldsymbol{\tau}.$$
$$\text{Define } [b_1, b_2, b_3]^T = \operatorname{diag}\left(\mathbf{M}_0^{-1}\right) \text{ and } \mathbf{e} = \mathbf{q}_d - \mathbf{q}, \quad (21)$$

where $\mathbf{q}_d = [q_{d1} \;\; q_{d2} \;\; q_{d3}]^T$ is the desired trajectory.

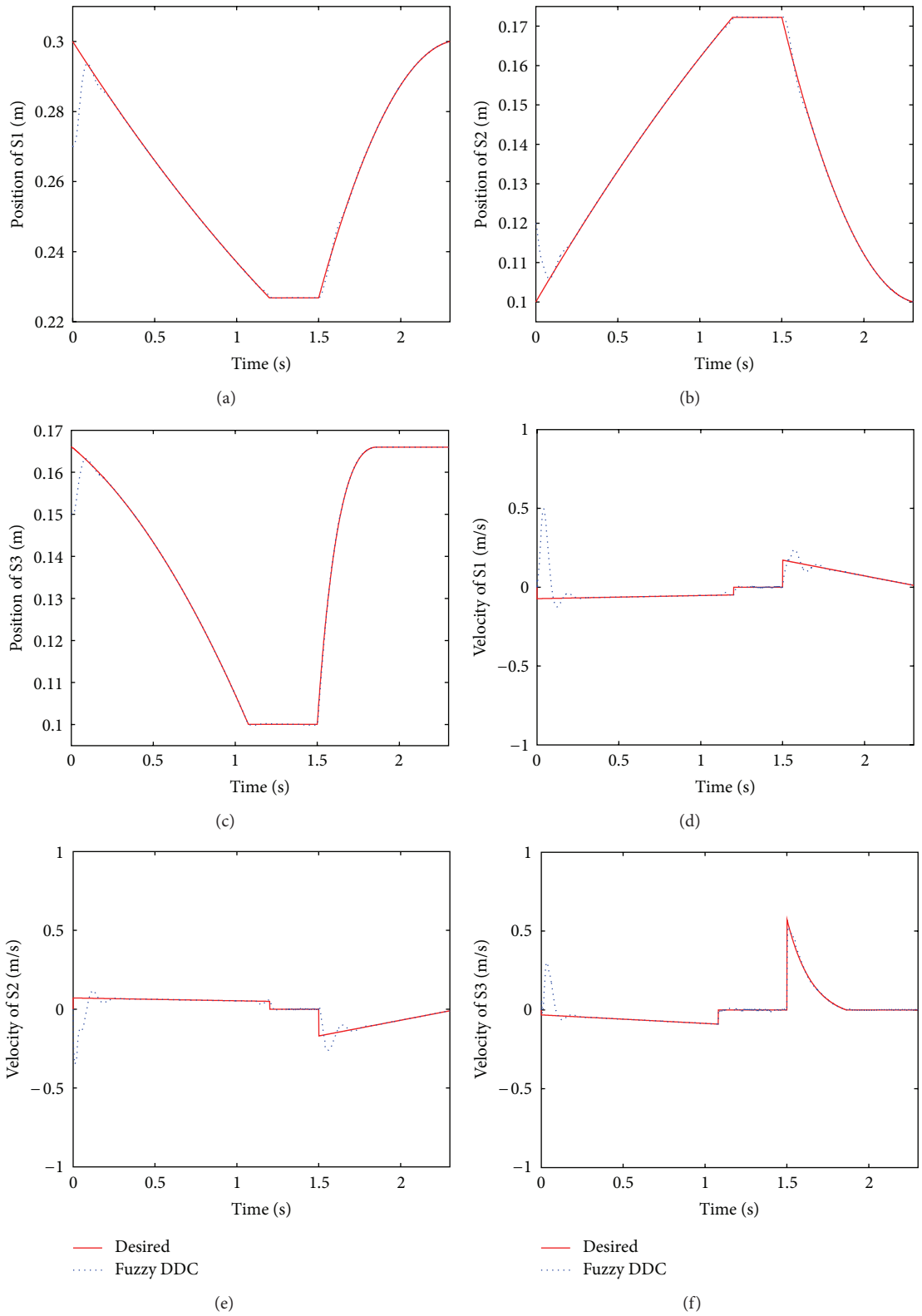


FIGURE 10: Responses to desired throwing trajectories under DDC. (a) Position responses of S1, (b) position responses of S2, (c) position responses of S3, (d) velocity responses of S1, (e) velocity responses of S2, and (f) velocity responses of S3.

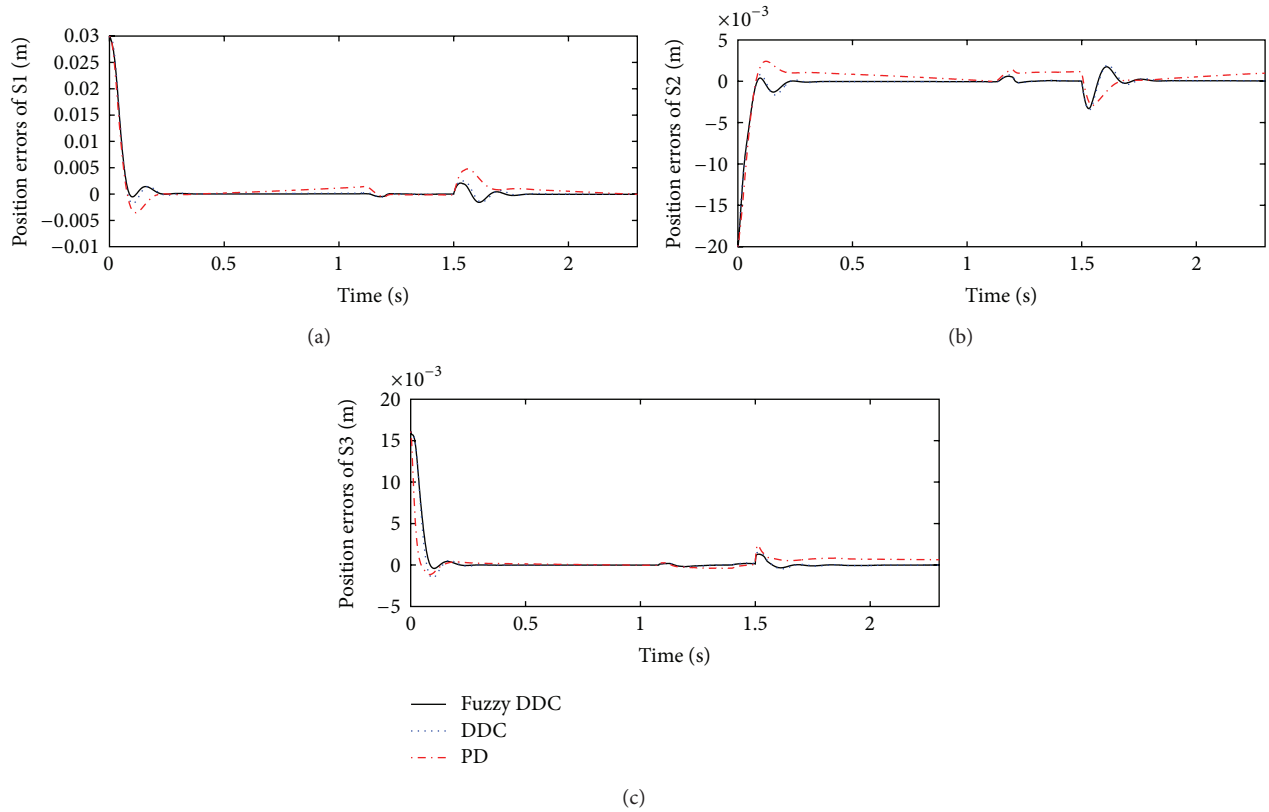


Figure 11: Comparison of position tracking errors. (a) Tracking error of S1, (b) tracking error of S2, and (c) tracking error of S3.

The HHA system can be formed by a set of coupled equations with predetermined input-output parings:

$$\begin{aligned} \ddot{q}_1 &= p_1\left(\dot{\mathbf{q}}, \mathbf{q}, \boldsymbol{\tau}, \tau_{d1}, \mathbf{F}_1\right) + b_1\tau_1, \\ \ddot{q}_2 &= p_2\left(\dot{\mathbf{q}}, \mathbf{q}, \boldsymbol{\tau}, \tau_{d2}, \mathbf{F}_2\right) + b_2\tau_2, \\ \ddot{q}_3 &= p_3\left(\dot{\mathbf{q}}, \mathbf{q}, \boldsymbol{\tau}, \tau_{d3}, \mathbf{F}_3\right) + b_3\tau_3. \end{aligned} \tag{22}$$

Define the total disturbance of each loop as

$$f_i = p_i\left(\dot{\mathbf{q}}, \mathbf{q}, \boldsymbol{\tau}, \tau_{di}, \mathbf{F}_i\right) + \left(b_i - b_{i0}\right)\tau_i, \quad i = 1, 2, 3, \tag{23}$$

where b_{i0} is the approximate value of b_i, f_i represents the combined effect of internal dynamics and external disturbances in the ith loop, including the cross-channel interference. Then (22) can be written as

$$\begin{aligned} \ddot{q}_1 &= f_1 + b_{10}\tau_1, \\ \ddot{q}_2 &= f_2 + b_{20}\tau_2, \\ \ddot{q}_3 &= f_3 + b_{30}\tau_3. \end{aligned} \tag{24}$$

Now, the HHA system has been decoupled to a three-loop system as shown in Figure 5. Then an ADRC based SISO controller can be designed for each loop independently. The task of ADRC comes down to a critical subtask: estimate f_i in real time. The estimation problem of f_i leads us to a unique state observer known as the extended state observer (ESO).

Considering the first loop, the structure of DDC in first loop is given in Figure 6.

Let $x_{11} = q_1$, $x_{12} = \dot{q}_1$, $x_{13} = f_1$, which is added as an extended state. Assume f_1 is differentiable, and $\dot{f}_1 = h$. The augmented model of (24) is

$$\begin{aligned} \dot{x}_{11} &= x_{12}, \qquad \dot{x}_{12} = x_{13} + b_{10}\tau_1, \\ \dot{x}_{13} &= h_1, \qquad q_1 = x_{11}. \end{aligned} \tag{25}$$

Letting $\mathbf{x}_1 = \left[x_{11} \;\; x_{12} \;\; x_{13}\right]^T$, the augmented state space form of (25) is

$$\dot{\mathbf{x}}_1 = \mathbf{A}_1\mathbf{x} + \mathbf{B}_1\tau_1 + \mathbf{E}_1 h_1, \qquad q_1 = \mathbf{C}_1\mathbf{x}_1, \tag{26}$$

where

$$\begin{aligned} \mathbf{A}_1 &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \qquad \mathbf{B}_1 = \begin{bmatrix} 0 \\ b_{10} \\ 0 \end{bmatrix}, \\ \mathbf{C}_1 &= \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T, \qquad \mathbf{E}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \end{aligned} \tag{27}$$

The state observer denoted as the linear extended state observer (LESO) of (26) is constructed as

$$\dot{\mathbf{z}}_1 = \mathbf{A}_1 + \mathbf{B}_1\tau_1 + \mathbf{L}_1\left(q_1 - z_{11}\right), \qquad \hat{q}_1 = \mathbf{C}\mathbf{z}_1, \tag{28}$$

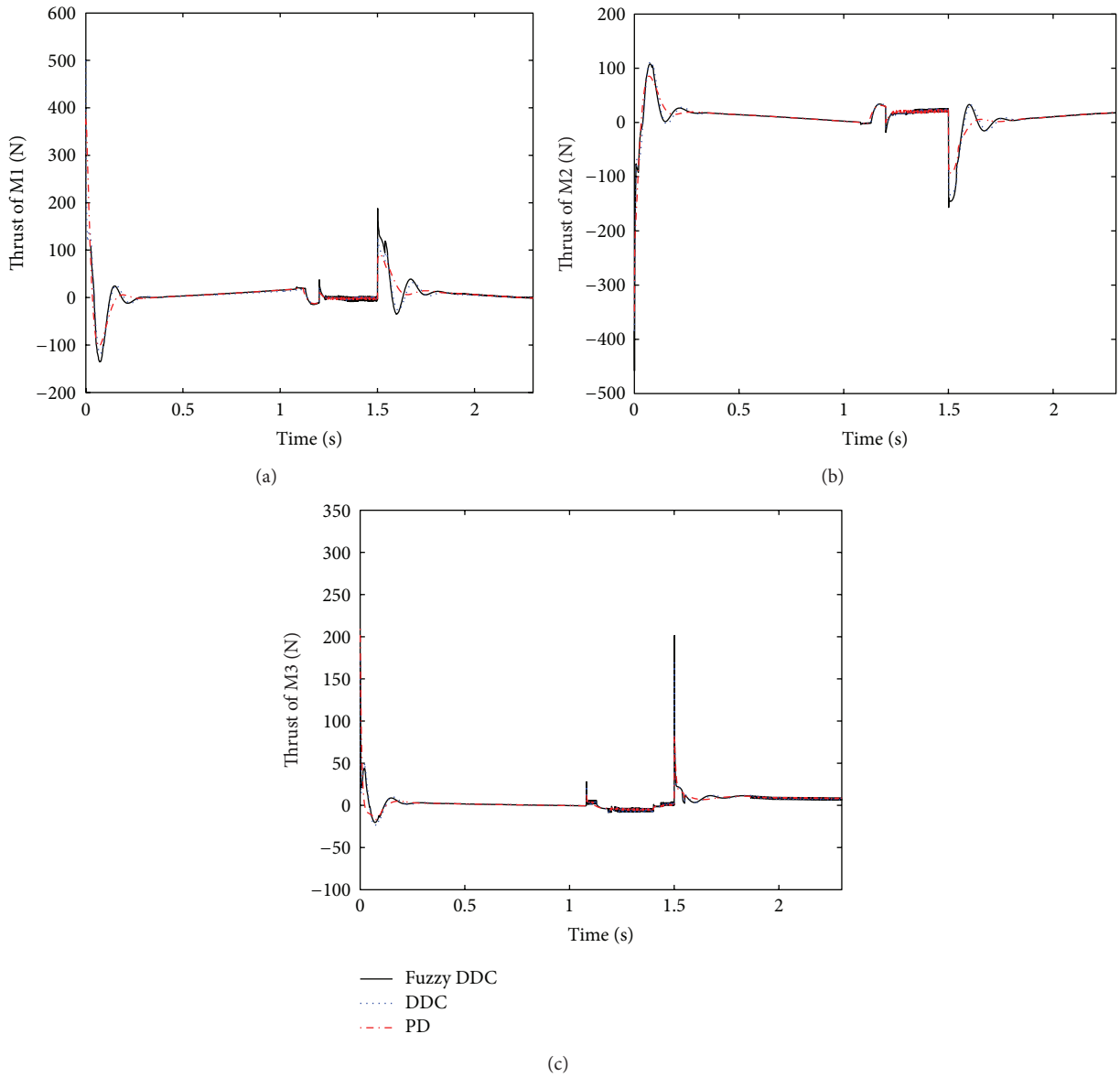


Figure 12: Comparison of input thrust. (a) Input thrust of M1, (b) input thrust of M2, and (c) input thrust of M3.

where $\mathbf{L}_1$ is the observer gain vector:

$$\mathbf{L}_1 = [\beta_{11} \quad \beta_{12} \quad \beta_{13}]^T. \tag{29}$$

With the state observer properly designed, z_{11}, z_{12}, and z_{13} closely track q_1, $\dot{q}_1$, and f_1, respectively. The controller is given by

$$\tau_1 = \frac{-z_{13} + \tau_{10}}{b_{10}}. \tag{30}$$

Ignoring the estimation error in z_{13} to f_1, the plant is reduced to a unit gain double integrator,

$$\ddot{q}_1 = (f_1 - z_{13}) + \tau_{10} \approx \tau_{10}, \tag{31}$$

which can be easily controlled by a linear state error feedback (LSEF) controller:

$$\tau_{10} = k_{p1}(q_{d1} - z_{11}) + k_{d1}(\dot{q}_{d1} - z_{12}), \tag{32}$$

where q_{d1} is the desired trajectory of S1. Then the total control law of first loop is found as

$$\tau_1 = \frac{k_{p1}(q_{d1} - z_{11}) + k_{d1}(\dot{q}_{d1} - z_{12}) - z_{13}}{b_{10}}. \tag{33}$$

In order to simplify the tuning of ADRC parameters, a parameter tuning method of the LESO is proposed in [24]; with using the pole placement technique, all the observer poles are at $-\omega_{01}$, making ω_{01} the only tuning parameter for the LESO. That is

$$\mathbf{L}_1 = [\beta_{11} \quad \beta_{12} \quad \beta_{13}]^T = [3\omega_{01} \quad 3\omega_{01}^2 \quad \omega_{01}^3], \tag{34}$$

where ω_{01} is the bandwidth of the LESO 1. This makes LADRC only have three tuning parameters ω_{01}, k_{p1}, and k_{d1}, compared to traditional ADRC controller.

4.2. Stability

Theorem 1. *Assuming h_1 is bounded, the LESO is bounded-input bounded-output (BIBO) stable.*

Proof. Let $e_{1,i} = x_{1,i} - z_{1,i}$, $i = 1, 2, 3$, and combine (28) and (29) and subtract it from (26); the error equation can be written as

$$\dot{\mathbf{e}}_1 = \mathbf{A}_{e1}\mathbf{e}_1 + \mathbf{E}_1 h_1, \tag{35}$$

where

$$\mathbf{A}_{e1} = \mathbf{A}_1 - \mathbf{L}_1\mathbf{C}_1 = \begin{bmatrix} -3\omega_{01} & 1 & 0 \\ -3\omega_{01}^2 & 0 & 1 \\ -\omega_{01}^3 & 0 & 0 \end{bmatrix}, \tag{36}$$

and $\mathbf{E}$ is defined in (26). The characteristic polynomial of $\mathbf{A}_{e1}$ is

$$\lambda_1(s) = s^3 + 3\omega_{01}s^2 + 3\omega_{01}^2 s + \omega_{01}^3. \tag{37}$$

Letting $\omega_{01} > 0$, it is obvious the roots of the characteristic polynomial of $\mathbf{A}_{e1}$ are all in the left plane and h_1 is bounded; we can draw the conclusion that the LESO 1 is bounded-input bounded-output (BIBO) stable. □

Theorem 2. *Assuming h_1 and q_{d1} are bounded, the DDC 1 design from* (28) *to* (33) *yields a BIBO stable closed-loop system.*

Proof. From (30) and (33), a state feedback equation can be described as

$$\tau_1 = \frac{k_{p1}(q_{d1} - z_{11}) + k_{d1}(\dot{q}_{d1} - z_{12}) - z_{13}}{b_{01}} = F_1\mathbf{z}_1 + R_1, \tag{38}$$

where

$$F_1 = \left(\frac{1}{b_{01}}\right)\begin{bmatrix} -k_{p1} & -k_{d1} & -1 \end{bmatrix}, \qquad R_1 = \left(\frac{1}{b_{01}}\right)\left(k_{p1}q_{d1} + k_{d1}\dot{q}_{d1}\right). \tag{39}$$

Then the closed-loop system can be transformed into the following state equation:

$$\begin{bmatrix} \dot{\mathbf{x}}_1 \\ \dot{\mathbf{z}}_1 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{B}_1\mathbf{F}_1 \\ \mathbf{L}_1\mathbf{C}_1 & \mathbf{A}_1 - \mathbf{L}_1\mathbf{C}_1 + \mathbf{B}_1\mathbf{F}_1 \end{bmatrix}\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{z}_1 \end{bmatrix} + \begin{bmatrix} \mathbf{B}R_1 + \mathbf{E}_1 h_1 \\ R_1 \end{bmatrix}. \tag{40}$$

By applying elementary row and column operations, it is obvious that the closed-loop eigenvalues satisfy

$$\begin{aligned} &\operatorname{eig}\left(\begin{bmatrix} \mathbf{A}_1 & \mathbf{B}_1\mathbf{F}_1 \\ \mathbf{L}_1\mathbf{C}_1 & \mathbf{A}_1 - \mathbf{L}_1\mathbf{C}_1 + \mathbf{B}_1\mathbf{F}_1 \end{bmatrix}\right) \\ &\quad = \operatorname{eig}\left(\begin{bmatrix} \mathbf{A}_1 + \mathbf{B}_1\mathbf{F}_1 & \mathbf{B}_1\mathbf{F}_1 \\ \mathbf{0} & \mathbf{A}_1 - \mathbf{L}_1\mathbf{C}_1 \end{bmatrix}\right) \\ &\quad = \operatorname{eig}(\mathbf{A}_1 + \mathbf{B}_1\mathbf{F}_1) \cup \operatorname{eig}(\mathbf{A}_1 - \mathbf{L}_1\mathbf{C}_1) \\ &\quad = \left\{\text{roots of } s^2 + k_{d1}s + k_{p1}\right\} \\ &\qquad \cup \left\{\text{roots of } s^3 + 3\omega_{01}s^2 + 3\omega_{01}^2 s + \omega_{01}^3\right\}. \end{aligned} \tag{41}$$

Letting $k_{d1}, k_{p1}, \omega_{01} > 0$, it is obvious the roots of $s^2+k_{d1}s+k_{p1}$ and $s^3+3\omega_{01}s^2+3\omega_{01}^2 s+\omega_{01}^3$ are all in the left plane; since h_1 and q_{d1} are bounded, we can draw the conclusion that the DDC 1 is bounded-input bounded-output (BIBO) stable.

Consulting the disturbance decoupling control design strategy and stability analysis in the first loop, it is easy to develop two disturbance decoupling controllers for two other loops, the two other control laws τ_2 and τ_3,

$$\tau_2 = \frac{k_{p2}(q_{d2} - z_{21}) + k_{d2}(\dot{q}_{d2} - z_{22}) - z_{23}}{b_{20}}, \qquad \tau_3 = \frac{k_{p3}(q_{d3} - z_{31}) + k_{d3}(\dot{q}_{d3} - z_{32}) - z_{33}}{b_{30}}. \tag{42}$$

□

4.3. Fuzzy Self-Tuning Disturbance Decoupling Control (FSDDC) for HHA. In this section, the fuzzy self-tuning disturbance decoupling control (FSDDC) with varying control gain is presented. The general structure of the proposed controller is given in Figure 7. The control gains of LSEF are changed dynamically by using the fuzzy logic unit in order to improve the performance of the controller; then the parameter regulation rules are given as follows:

$$k_{pi} = k_{pi0} + \xi_i \Delta k_{pi}, \quad i = 1, 2, 3, \qquad k_{di} = k_{di0} + \eta_i \Delta k_{di}, \quad i = 1, 2, 3, \tag{43}$$

where k_{pi0} and k_{di0} are the initial gains of LSEF, Δk_{pi} and Δk_{di} are the output of fuzzy controllers, and ξ_i and η_i are the scale factors of FSDDC.

As shown in Figure 8, the membership functions are used for the fuzzification of the inputs, which are error e_i and derivative of error $\dot{e}_i$. Here NB, NM, NS, ZO, PS, PM, and PB stand for negative big, negative medium, negative small, zero, positive small, positive medium, and positive big, respectively. The outputs are Δk_{pi} and Δk_{di}; then the initial control gains will be adjusted by the outputs and scale factors as shown in (43). This gives the ability to make the control gain value of the proposed controller smaller or bigger than the gain of the conventional DDC whenever necessary.

The rule tables for Δk_{pi} and Δk_{di} are constructed and given in Tables 5 and 6.

Since ξ_i and η_i are chosen to guarantee that the control gains are always kept positive during the fuzzy adaptation, the BIBO stability of the controlled system is preserved.

5. Simulation

A set of numerical simulations is used here to verify the effectiveness of the proposed DDC and FSDDC. In addition, classical PD controller, which is commonly used for control of robot manipulators in industry because of its simple structure, is also applied to the HHA used for comparison. The numerical values of parameters of the HHA are listed in Table 7.

The reference signals are given in Section 2.2. The initial values of the system are selected as

$$\begin{aligned} x_1(0) &= 0.27, \qquad x_2(0) = 0.12, \qquad x_3(0) = 0.15, \\ \dot{x}_1(0) &= 0, \qquad \dot{x}_2(0) = 0, \qquad \dot{x}_3(0) = 0. \end{aligned} \tag{44}$$

In order to check the robustness of the proposed controllers, a load is introduced and it will be put on the hand in 1.4 seconds to 1.55 seconds; the variation of the mass of the wrist is given in Figure 9.

Normally, the distributed noise and the parameter uncertainties of the system always exist, which can be thought as disturbing torques acting on the joints, and are assumed to be time varying as

$$\tau_d = \begin{bmatrix} 20\dot{x}_1 + 20\sin(x_1) + 5\,\mathrm{sgn}(\dot{x}_1) \\ 20\dot{x}_2 + 20\sin(x_2) + 5\,\mathrm{sgn}(\dot{x}_2) \\ 15\dot{x}_3 + 15\sin(x_3) + 3\,\mathrm{sgn}(\dot{x}_3) \end{bmatrix}. \tag{45}$$

In the following passage, our proposed FSDDC and DDC are used for stabilizing the HHA system in comparison with the PD controller, with the gains of the FSDDC, DDC, and PD controllers being given in Tables 8, 9,and 10.

The dynamic tracking performances of the proposed FSDDC are illustrated in Figure 10, and one can find that the proposed FSDDC provides a reasonable tracking capability in the various disturbances and uncertainties. The mass variation applied after 1.4 seconds almost does not affect the tracking performance.

Figure 11 illustrates the position tracking errors. For a clear comparison, the requested input thrusts of the two controllers are shown in Figure 12.

It is easy to see that the DDC has better tracking performance than the PD controller; the proposed DDC has the smaller tracking error and more convergence to zero than the PD controller; the performance of DDC is further improved by combination with the FLC; the FSDDC has smaller overshoot and shorter setup time than DDC.

The requested input thrust of the proposed FSDDC and DDC is slightly bigger than that of the PD controller, which are all in the range of output thrust of linear motors, and proves the efficiency of the proposed controller.

As a result, we can conclude that the proposed DDC and FSDDC can give a better tracking over the commonly used PD control. In this situation, the HHA can complete the throwing process as requested.

6. Conclusion

This paper addressed the robust trajectory tracking problem for a serial-parallel hybrid humanoid arm (HHA) throwing process in the presence of uncertainties and disturbances. The FSDDC control, with the advantages of both DDC and FLC, has shown a significant improvement over the PID control under the same conditions. The FSDDC can readily decouple the multi-input and multioutput processes by taking the interactions among different parts of the process as disturbances to be estimated and cancelled; it has strong disturbance rejection ability and robustness to uncertainties, which widely exist in throwing processes. The effectiveness of the designed unique dynamic FSDDC strategy was illustrated by simulation examples. We expect that, as we continue to work with the implementation, we may improve the response even further.

References

[1] W. B. J. Hakvoort, R. G. K. M. Aarts, J. van Dijk, and J. B. Jonker, "Lifted system iterative learning control applied to an industrial robot," *Control Engineering Practice*, vol. 16, no. 4, pp. 377–391, 2008.

[2] S. Wei, Y. Wang, and Y. Zuo, "Wavelet neural networks robust control of farm transmission line deicing robot manipulators," *Computer Standards and Interfaces*, vol. 34, no. 3, pp. 327–333, 2012.

[3] J. P. Merlet, *Parallel Robots*, Kluwer Academic, Dordecht, The Netherlands, 2000.

[4] Y. Pi and X. Wang, "Trajectory tracking control of a 6-DOF hydraulic parallel robot manipulator with uncertain load disturbances," *Control Engineering Practice*, vol. 19, no. 2, pp. 185–193, 2011.

[5] C. K. Lin, "H_∞ reinforcement learning control of robot manipulators using fuzzy wavelet networks," *Fuzzy Sets and Systems*, vol. 160, no. 12, pp. 1765–1786, 2009.

[6] N. Yagiz and Y. Hacioglu, "Robust control of a spatial robot using fuzzy sliding modes," *Mathematical and Computer Modelling*, vol. 49, no. 1-2, pp. 114–127, 2009.

[7] T. H. S. Li and Y. C. Huang, "MIMO adaptive fuzzy terminal sliding-mode controller for robotic manipulators," *Information Sciences*, vol. 180, no. 23, pp. 4641–4660, 2010.

[8] T. Sun, H. Pei, Y. Pan, H. Zhou, and C. Zhang, "Neural network-based sliding mode adaptive control for robot manipulators," *Neurocomputing*, vol. 74, no. 14, pp. 2377–2384, 2011.

[9] S. Yu, X. Yu, B. Shirinzadeh, and Z. Man, "Continuous finite-time control for robotic manipulators with terminal sliding mode," *Automatica*, vol. 41, no. 11, pp. 1957–1964, 2005.

[10] M. B. R. Neila and D. Tarak, "Adaptive terminal sliding mode control for rigid robotic manipulators," *International Journal of Automation and Computing*, vol. 8, no. 2, pp. 215–220, 2011.

[11] Y. Chen and J. E. McInroy, "Decoupled control of flexure-jointed hexapods using estimated joint-space mass-inertia matrix," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 3, pp. 413–421, 2004.

[12] C. Yang, Q. Huang, and J. Han, "Computed force and velocity control for spatial multi-DOF electro-hydraulic parallel manipulator," *Mechatronics*, vol. 22, pp. 715–722, 2012.

[13] C. Yang, Q. Huang, and J. Han, "Decoupling control for spatial six-degree-of-freedom electro-hydraulic parallel robot,"

Robotics and Computer-Integrated Manufacturing, vol. 28, no. 1, pp. 14–23, 2012.

[14] N. Minorsk, "Directional stability of automatically steered bodies," *Journal of ASNE*, vol. 42, no. 2, pp. 280–309, 1922.

[15] K. J. Astrom and H. Tore, *Advanced PID Control*, The Instrumentation, Systems, and Automation Society, Triangle Park, NC, USA, 2005.

[16] Y. Wang and Z. Jin, "Dynamics modeling and robust trajectory tracking control for a class of hybrid humanoid arm based on neural network," *Chinese Journal of Mechanical Engineering*, vol. 22, no. 3, pp. 355–363, 2009.

[17] Q. Zheng, Z. Chen, and Z. Gao, "A practical approach to disturbance decoupling control," *Control Engineering Practice*, vol. 17, no. 9, pp. 1016–1025, 2009.

[18] J. Han, "Auto-disturbance rejection control and its applications," *Control and Decision*, vol. 13, no. 1, pp. 19–23, 1998 (Chinese).

[19] J. Han, "Nonlinear design methods for control systems," in *Proceedings of the 14th International Federation of Accountants IFAC World Congress*, pp. 521–526, 1999.

[20] Z. Gao, Y. Huang, and J. Han, "An alternative paradigm for control system design," in *Proceedings of the 40th IEEE Conference on Decision and Control (CDC '01)*, pp. 4578–4585, Orlando, Fla, USA, December 2001.

[21] Z. Gao, "Scaling and Bandwidth-parameterization based controller tuning," in *Proceedings of the 2003 American Control Conference*, vol. 6, pp. 4989–4996, June 2003.

[22] G. Tian and Z. Gao, "Frequency response analysis of active disturbance rejection based control system," in *Proceedings of the 16th IEEE International Conference on Control Applications (CCA '07)*, pp. 1595–1599, October 2007.

[23] J. Han, "A class of extended state observers for uncertain systems," *Control and Decision*, vol. 10, no. 1, pp. 85–88, 1995 (Chinese).

[24] Q. Zheng, L. Q. Gao, and Z. Gao, "On validation of extended state observer through analysis and experimentation," *Journal of Dynamic Systems, Measurement and Control*, vol. 134, no. 2, Article ID 024505, 2012.

[25] Z. Gao, "Active disturbance rejection control: a paradigm shift in feedback control system design," in *Proceedings of the 2006 American Control Conference*, pp. 2399–2405, June 2006.

[26] J. Han, "From PID to active disturbance rejection control," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 3, pp. 900–906, 2009.

[27] Y. Huang, K. Xu, J. Han, and J. Lam, "Flight control design using extended state observer and non-smooth feedback," in *Proceedings of the 40th IEEE Conference on Decision and Control (CDC '01)*, pp. 223–228, December 2001.

[28] J. Vincent, D. Morris, N. Usher et al., "On active disturbance rejection based control design for superconducting RF cavities," *Nuclear Instruments and Methods in Physics Research A*, vol. 643, no. 1, pp. 11–16, 2011.

[29] G. K. I. Mann, B. G. Hu, and R. G. Gosine, "Analysis of direct action fuzzy PID controller structures," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 29, no. 3, pp. 371–388, 1999.

[30] J. Carvajal, G. Chen, and H. Ogmen, "Fuzzy PID controller: design, performance evaluation, and stability analysis," *Information Sciences*, vol. 123, no. 3, pp. 249–270, 2000.

[31] H. B. Kazemian, "Comparative study of a learning fuzzy PID controller and a self-tuning controller," *ISA Transactions*, vol. 40, no. 3, pp. 245–253, 2001.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 149562, 10 pages
http://dx.doi.org/10.1155/2013/149562

Research Article

Fault Prognostic Based on Hybrid Method of State Judgment and Regression

Xiaobin Li,[1,2] Jiansheng Qian,[1] and Gai-ge Wang[2]

[1] *School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221008, China*
[2] *School of Computer Science and Technology, Jiangsu Normal University, Xuzhou 221116, China*

Correspondence should be addressed to Jiansheng Qian; qianjsh@cumt.edu.cn

Received 19 July 2013; Accepted 29 September 2013

Academic Editor: Amir H. Gandomi

Fault prognostic is one of the most important problems in equipment health management system. This paper presents a hybrid method of mixture of Gaussian hidden Markov model (MG-HMM) and fixed size least squares support vector regression (FS-LSSVR) for fault prognostic. The system is established based on three parts. The first part trains the MG-HMM and FS-LSSVR model. According to the known samples, several MG-HMM models can be learned based on expectation maximization (EM) algorithm. Then, the forward variables can be calculated based on these MG-HMM models. Based on these forward variables, the corresponding FS-LSSVR models are built. All the MG-HMM models and corresponding FS-LSSVR models are combined into a model library. The second part recognizes the unknown sample based on the model library. This part obtains the MG-HMM model and FS-LSSVR model by maximization likelihood calculation between the unknown sample and MG-HMM models. The third part of the system calculates the forward variables based on the MG-HMM obtained from the second part. These forward variables are inputted into the corresponding FS-LSSVR model to compute the remaining useful life (RUL) of the unknown sample. Finally, we carry out experiments on benchmark data set to verify the proposed method. The results illustrate the effectiveness of the hybrid method.

1. Introduction

RUL is one of the most important problems in many application areas such as condition based maintenance (CBM) [1], fault prognostics (FP) [2], and prognostics and health management (PHM) system [3]. Obviously, the RUL of a system or a component is a random variable. Three techniques are applied to estimate the RUL: life cycle model, expert knowledge system, and data-driven model [4]. The data-driven model compromises the merits of adaptability, low cost, and little expert knowledge. For these reasons, it has been widely concerned in recent years.

Many artificial intelligence (AI) techniques have been applied to fault prognostic. Expert system (ES) is a computer system which consists of a knowledge database and an inference mechanism. The knowledge database contains domain knowledge for the solution of problems. This approach is not best for situations where there is a lack of expert knowledge [5]. ES cannot handle new situations not covered explicitly in its knowledge database. Fuzzy logic can represent uncertainty of complexity system while it is no feasible where there is no suitable membership function [6]. Artificial Neural Network (ANN) can simulate the structures and functions of biological neural networks. This approach can capture potential knowledge from input patterns. However there is no standard method to determine the structure of the network [7]. Support vector machine (SVM) project feature space into a higher dimensional space by a kernel function and finds an optimized separation surface in the higher dimensional space. This approach can achieve better generalization capability than ANN while there this no standard kernel function selection method for SVM [8].

Hidden Markov model (HMM) has demonstrated its superior performance in lots of application areas, such as speech recognition and gesture recognition. It is more suitable for modeling on stationary stochastic signal. Meanwhile,

HMM is also inspected and applied into fault diagnosis or fault prognostics [9]. Comparing with many other RUL estimation methods, a distinct advantage of the HMM model is that HMM model can give an intelligible explanation. Bunks et al. firstly put forward a HMM model built on the Westland helicopter gear monitoring data, and then the RUL of gear is estimated based on the HMM model [10]. In this paper, the authors estimate each of the 68 operating conditions with a different eight-dimensional Gaussian model. At the same time, they make hypothesis that the health state could not revert once it departs from the current health state during its operation. Baruah and Chinnam apply HMM to model the metal cutting tools [11, 12]. The proposed method can simultaneously meet the requirements of fault diagnosis and fault prognostics. To describe a three-state transition procedure, a two-mixture Gaussian model is used for modeling. Based on the previous model, once the previous state transition time is known, the next state transition time can be estimated. Camci and Chinnam make a thorough inspection on drill bit health evaluation [13]. They define the drill bit health level as drilling holes number. Each drilling history time series can be used to train a separate HMM model. These HMM models are used to estimate the drill bit health in time. Then a hierarchy HMM is established to model the state transition relation in which the upper HMM model describes the drill bit health state transition relationship while the lower HMM model describes the drilling time series emitted from one state in the upper HMM model. Once the drill bit health is determined, the RUL is calculated based on the probability transition matrix in the upper HMM model. Ocak et al. make a decomposition on the bearing vibration signals by using wavelet packet technique. The Nth layer node data is used as feature vector for HMM training. If the bearing state need to be judged, the HMM models will be applied into likelihood calculation. The experimental results found that, as the bearing wears, the likelihood gradually drops [14]. Tai et al. apply HMM model into condition monitoring of nozzle [15]. Zhou et al. combine the HMM model and belief rule base under a variable circumstance, where belief rules are used to model the dynamic environment. Eventually, fault diagnosis and fault prognostics are implemented for the complex system based on this mixture model [16]. Tobon-Mejia et al. establish the models on the bearing data with mixture of gaussian hidden Markov model (MG-HMM). Once a sequence classification is determined, a graph-based path traversal algorithm is carried on to estimate the RUL of bearing [17]. Liu et al. present a hybrid method of HMM and LSSVR to predict the RUL. The LSSVR is used to predict the future features based on the past samples and the HMM is used to judge the future state according to the future predicted features [18].

Motivated by the above approaches for fault prognostic, in this paper, we present a hybrid method of state judgment and regression. The state judgment is implemented by MG-HMM model. The MG-HMM not only can make condition recognition but also provides detailed health indexes for unknown sample. The regression is implemented by FS-LSSVR. FS-LSSVR is a version of SVM which is more suitable for large scale application. The RUL can be calculated by the FS-LSSVR model built on the health indexes obtained from MG-HMM. We also compare the RUL prediction performance between artificial neural network (ANN) and FS-LSSVR. The experimental results show that the hybrid method is effective for fault prognostic.

This paper is organized as follows. Section 2 introduces the MG-HMM and its training. Section 3 introduces the FS-LSSVR model which is fit for large scale problem. Section 4 gives a detailed description for left-right HMM (LR-HMM) and system architecture which is used for fault diagnosis and fault prognostic. The experiments are carried out on a benchmark dataset to illustrate the effectiveness of the HMM training method in Section 5. The last section is the conclusion of this paper.

2. HMM Model and Its Training

2.1. HMM Model. Discrete hidden Markov model (DHMM) is a doubly stochastic process [19], and a DHMM model usually contains the following elements:

(1) $S = \{s_1, s_2, \ldots, s_N\}$, which is a finite set of states where each element means a distinct state,

(2) $V = \{v_1, v_2, \ldots, v_M\}$, which is a set of output symbols,

(3) A, which is state transition probability matrix where $a_{ij} = P[q_{t+1} = s_j \mid q_t = s_i]$, $1 \le i, j \le N$,

(4) $B = \{b_j(k)\}$, which is an observation value probability distribution, where $b_j(k) = P[v_k \mid q_t = s_j]$, $1 \le j \le N$, $1 \le k \le M$,

(5) $\pi = \{\pi_i\}$, which is an initial state probability distribution, where each element means a probability of the initial state. $\pi_i = P(q_1 = s_i)$, $1 \le i \le N$.

Usually, a more compact model $\lambda = (A, B, \pi)$ is used to represent a HMM model for there is an implication definition N and M in A and B.

DHMM only considers the case when the observations were discrete symbols chosen from a finite alphabet. There are some disadvantages for this method if the observations are continuous signals. Although it is possible to quantize such continuous signals via codebooks and so forth, there might be serious degradation associated with such quantization. Hence, it would be advantageous to use MG-HMM with continuous observation densities.

The continuous observation under a special state can be described as a probability density function. Usually, mixture Gaussian distribution probability density function is used to describe the probability density function:

$$b_j(O) = \sum_{m=1}^{M} c_{jm} N\left(O, \mu_{jm}, U_{jm}\right), \quad 1 \le j \le N, \tag{1}$$

where O is the vector being modeled, c_{jm} is the mixture coefficient for the mth mixture in state j, and N is a Gaussian distribution function, with mean vector μ_{jm} and covariance

matrix U_{jm} for the mth mixture component in state j. The mixture coefficient c_{jm} satisfies the stochastic constraint

$$\sum_{m=1}^{M} c_{jm} = 1, \quad 1 \le j \le N,$$
$$c_{jm} \ge 0, \quad 1 \le j \le N, \quad 1 \le m \le M. \tag{2}$$

So that the probability density function is properly normalized; that is,

$$\int_{-\infty}^{\infty} b_j(x)\, dx = 1, \quad 1 \le j \le N. \tag{3}$$

2.2. Model Training. DHMM parameter estimation problem can be defined as how to adjust unknown model parameters $\lambda = (A, B, \pi)$ to maximize $P(O \mid \lambda)$.

Baum-Welch (BW) or gradient descent algorithm can be applied to solve this problem. Now, a brief description of BW algorithm is provided here.

Firstly, a variable $\xi_t(i, j) = P(q_t = s_i, q_{t+1} = s_j \mid O, \lambda)$ is defined to represent a transition probability from state s_i at time t to state s_j at time $t + 1$.

By definition, it can be deduced that

$$\xi_t(i, j) = \frac{\alpha_t(i)\, a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j)}{P(O \mid \lambda)} = \frac{\alpha_t(i)\, a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t(i)\, a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j)}. \tag{4}$$

Hence, a probability under state s_i at time t can be defined as

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i, j). \tag{5}$$

Therefore, the three parameters of HMM model can be estimated. The initial probability distribution can be calculated as

$$\overline{\pi_i} = \gamma_1(i). \tag{6}$$

The transition probability distribution can be calculated as

$$\overline{a_{ij}} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}. \tag{7}$$

The emission probability distribution can be calculated as

$$\overline{b_j(k)} = \frac{\sum_{t=1}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} \tag{8}$$

Given an observation sequence $O = O_1, O_2, \ldots O_T$, Baum-Welch method may be used to estimate λ such that $P(O \mid \lambda)$ is locally maximized. Firstly, three variables are defined as follows.

Forward variable $\alpha_t(i) = P(O_1, O_1, \ldots O_t, q_t = S_i \mid \lambda)$, that is, the probability of the partial observation sequence, $O_1, O_2, \ldots O_t$, and state S_i at time t, given the model λ.

Backward variable $\beta_t(i) = P(O_{t+1}, O_{t+2}, \ldots O_T \mid q_t = S_i, \lambda)$, that is, the probability of the partial observation sequence from $t + 1$ to the end, given state S_i at time t and model λ.

$\omega_t(i, j) = P(q_t = S_i, q_{t+1} = S_j \mid O, \lambda)$, the probability of being in state S_i at time t, and S_j at time $t + 1$, given the model λ and the observation sequence O.

The detailed model training steps of MG-HMM are listed as follows.

Step 1. Some variables are firstly defined: state number N, Gaussian mixture component number M, initial state distribution with random initialization, state transition probability distribution A with random initialization and the other parameters needed to construct observation symbol probability distribution in state j with random initialization. Of course the random initialization should be satisfied with the stochastic constraint.

Step 2. Calculate $\alpha_t(i)$. This variable can be solved by dynamic programming algorithm. The solving process is as follows:

$$\alpha_1(i) = \pi_i b_i(O_1), \quad 1 \le j \le N,$$
$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i)\, a_{ij}\right] b_j(O_{t+1}),$$
$$1 \le j \le N, \quad 1 \le t \le T - 1. \tag{9}$$

Step 3. Calculate $\beta_t(i)$. The solving process is as follows:

$$\beta_T(i) = 1, \quad 1 \le j \le N,$$
$$\beta_t(j) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j), \quad 1 \le j \le N,$$
$$t = T - 1, T - 2, \ldots, 1. \tag{10}$$

Step 4. Calculate $\omega_t(i, j) = P(q_t = S_i, q_{t+1} = S_j \mid O, \lambda)$ for each time t:

$$\omega_t(i, j) = \frac{\alpha_t(i)\, a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j)}{P(O \mid \lambda)} = \frac{\alpha_t(i)\, a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t(i)\, a_{ij} b_j(O_{t+1})\, \beta_{t+1}(j)} \tag{11}$$

Step 5. Calculate $\gamma_t(i) = \sum_{i=1}^{N} \omega_t(i, j)$ for each time t.

Step 6. Calculate updated initial probability distribution π and state transition probability distribution A:

$$\overline{\pi_i} = \gamma_1(i),$$
$$\overline{a_{ij}} = \frac{\sum_{t=1}^{T-1} \omega_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}. \tag{12}$$

Step 7. For each time t, state j and the kth component probability in mixture Gaussian distribution can be expressed as

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N}\alpha_t(j)\beta_t(j)}\right]\left[\frac{c_{jk}N\left(O_t,\mu_{jk},U_{jk}\right)}{\sum_{m=1}^{M}c_{jk}N\left(O_t,\mu_{jk},U_{jk}\right)}\right]. \tag{13}$$

Step 8. Estimate the parameters in observation probability density function:

$$\begin{aligned}\overline{c_{jk}} &= \frac{\sum_{t=1}^{T}\gamma_t(j,k)}{\sum_{t=1}^{T}\sum_{k=1}^{M}\gamma_t(j,k)},\\ \overline{\mu_{jk}} &= \frac{\sum_{t=1}^{T}\gamma_t(j,k)O_t}{\sum_{t=1}^{T}\gamma_t(j,k)},\\ \overline{U_{jk}} &= \frac{\sum_{t=1}^{T}\gamma_t(j,k)\left(O_t-\mu_{jk}\right)\left(O_t-\mu_{jk}\right)'}{\sum_{t=1}^{T}\gamma_t(j,k)}.\end{aligned} \tag{14}$$

Step 9. The procedure may end or jump to Step 2 according to the given threshold.

The above model training algorithm can only be applied for a single observation sequence. In practical application, in order to obtain a more reliable estimation of parameters, one has to use multiple observation sequences.

For given multiple sequences $O = [O^{(1)}, O^{(2)}, \ldots, O^{(z)}]$ and each sequence $O^{(z)} = [O_1^{(z)}, O_2^{(z)}, \ldots, O_{T_z}^{(z)}]$, the goal of model training is $P(\lambda \mid O) = \max_\lambda \prod_{z=1}^{z} P(O^{(z)} \mid \lambda)$. We can adaptively modify the above algorithm to confirm with the multiple sequences. First of all, the likelihood $P(O^{(z)} \mid \lambda) = \sum_{i=1}^{N}\alpha_T^z(i)$ of each single sequence is calculated. Hence, the reestimation of parameters should consider the contribution weight of each sequence.

3. FS-LSSVR

SVM has been widely used in many areas [20]. Least squares support vector machine (LSSVM) [21] is a least squares version of SVM for classification problems. The solution of LSSVM follows directly from solving a set of linear equations. Furthermore, the support values in LSSVM are proportional to the errors.

3.1. LSSVM. Given a sample set $\{y_k, x_k\}_{k=1}^{N}$ where $x_k \in R^n$ is the kth input features and $y_k \in R$ is the kth label, LSSVM can be described as follows [21]:

$$\begin{aligned}&\min_{w,b,e} \quad J = \frac{1}{2}w^T w + \frac{1}{2}\gamma\sum_{k=1}^{N}e_k^2\\ &\text{subject to} \quad y_k\left[w^T\phi(x_k)+b\right] = 1-e_k,\\ &\qquad k = 1,\ldots,N.\end{aligned} \tag{15}$$

The solution to the constrained optimization problem follows from the Lagrangian:

$$\begin{aligned}L(w,b,e;\alpha) &= J(w,b,e)\\ &\quad - \sum_{k=1}^{N}\alpha_k\left\{y_k \mid \left[w^T\phi(x_k)+b\right]-1+e_k\right\}.\end{aligned} \tag{16}$$

Here, the variable α_k is Lagrange multiplier. The problems can be solved by the following equation:

$$\begin{aligned}\frac{\partial L}{\partial w} &= 0 \longrightarrow w = \sum_{k=1}^{N}\alpha_k y_k\phi(x_k),\\ \frac{\partial L}{\partial b} &= 0 \longrightarrow \sum_{k=1}^{N}\alpha_k y_k = 0,\\ \frac{\partial L}{\partial e_k} &= 0 \longrightarrow \alpha_k = \gamma e_k, \quad k = 1,\ldots,N,\\ \frac{\partial L}{\partial \alpha_k} &= 0 \longrightarrow y_k\left[w^T\phi(x_k)+b\right]-1+e_k = 0,\\ &\qquad k = 1,\ldots,N.\end{aligned} \tag{17}$$

Similar to the standard SVM, the w and $\phi(x_i)$ do not need to be solved. A linear KKT system can be established in place of the two order optimization system to eliminate w and e:

$$\begin{bmatrix}0 & 1_v^T\\ 1_v & \Omega+\gamma^{-1}I_N\end{bmatrix}\begin{bmatrix}b\\ \alpha\end{bmatrix} = \begin{bmatrix}0\\ Y\end{bmatrix}. \tag{18}$$

In the above linear equation, there are $Y = [y_1;\ldots;y_N]$ and $1_v = [1;\ldots;1]$, $e = [e_1;\ldots;e_N]$, and $\alpha = [\alpha_1;\ldots;\alpha_N]$. Meanwhile, according to the Mercer permission condition, the kernel matrix $\Omega \in R^{N\times N}$ can be written as

$$\Omega_{ij} = \phi(x_i)^T\phi\left(x_j\right) = K\left(x_i,x_j\right). \tag{19}$$

The solution of classification problem is as follows:

$$y(x) = \operatorname{sign}\left[\sum_{i=1}^{N}\alpha_i y_i K(x,x_i)+b\right]. \tag{20}$$

3.2. FS-LSSVR. Solving the LS-SVM requires the resolution for all samples, which is practical when the input space dimension is larger than sample size. However, when the sample size is very large, it is impossible to solve these questions by traditional LS-SVM method [22]. For example, the benchmark data set used in Section 5 to finish fault prognostic contains about 200 samples, and the total size is above 20000. For this case, LS-SVR needs to be adjusted to fit for large scale problems. FS-LSSVR is presented to solve this problem [22]. It makes use of the NystrÖm approximation but estimates the model in the primal within the LS-SVM setting [23]. Instead of a random subset, a subset selection method based upon quadratic Renyi entropy was proposed.

(1) Nyström Approximation in Dual Space. Nonlinear map φ can be explicitly represented by eigenvalue decomposition of

kernel matrix Ω and kernel function $K(x, x_j)$. For probability density $p(x)$, there is

$$\int K\left(x, x_j\right) \varphi_i(x)\, p(x)\, dx = \lambda_i \varphi_i\left(x_j\right). \tag{21}$$

ϕ is represented as follows:

$$\phi = \left[\sqrt{\lambda_1}\varphi_1, \sqrt{\lambda_2}\varphi_2, \ldots, \sqrt{\lambda_{nh}}\varphi_{nh}\right]. \tag{22}$$

Here, the φ is eigenvalue function.

The above problem can be transformed to an eigenvalue problem on the sample data set:

$$\frac{1}{N}\sum_{k=1}^{N} K\left(x_k, x_j\right) u_i\left(x_k\right) = \lambda_i^{(s)} u_i\left(x_j\right). \tag{23}$$

Then, the eigenvalue λ_i and eigenvalue vector u_i can be approximated by the eigenvalue and eigenvalue vector:

$$\widehat{\lambda_i} = \frac{1}{N}\lambda_i^{(s)}, \qquad \widehat{\varphi_i} = \sqrt{N} u_i. \tag{24}$$

The nonlinear map is estimated as follows:

$$\widehat{\phi_i}(x) = \frac{N}{\sqrt{\lambda_i^{(s)}}} \sum_{k=1}^{N} u_{ki} K\left(x_k, x^{(v)}\right). \tag{25}$$

(2) Sparse Spproximation for Subset. Small proportion of all samples selected to approximate all samples is the core idea of FS-LSSVR algorithm. The standard of sample selection is of great importance. Entropy maximization is a well-defined criterion for subset selection. Renyi entropy can be considered as one of the criteria:

$$H_R = -\log \int p(x)^2 dx. \tag{26}$$

It can be approximated as

$$\int \widehat{p}(x)^2 dx = \frac{1}{N^2} \mathbf{1}^{\mathbf{T}} \mathbf{\Omega} \mathbf{1}. \tag{27}$$

(3) Based on the analysis in (1) and (2), the algorithm can be listed as follows.

Step 1. Randomly select a subset with M samples from the original data set $\{y_k, x_k\}_{k=1}^{N}$. The two-order Renyi entropy is calculated for the selected samples. Then, the subset with maximization Renyi entropy is determined.

Step 2. A small kernel matrix Ω_M is constructed based on the selected samples.

Step 3. Compute the eigenvalue $\lambda_i^{(s)}$ and eigenvalue vector u_i on Ω_M.

Step 4. Nonlinear map $\widehat{\phi}(\mathrm{x_i})$ is calculated for all samples ($i = 1, \ldots, N$).

Step 5. The final regression problem is solved based on the above steps.

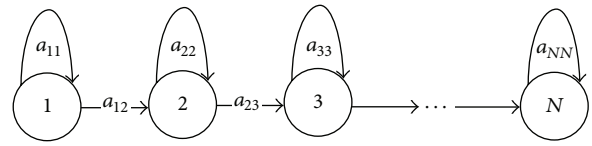


Figure 1: LR-HMM Model for RUL prediction.

4. System Framework for RUL Prediction

4.1. LR-HMM. Here LR-HMM model [7] is proposed for RUL prediction as shown in Figure 1.

The health status of system discussed here is $S = \{1, 2, 3, \ldots, N\}$. The health status is a oneway and irreversible. From Section 2, we know that the forward variable $\alpha_t(i) = P(O_1, O_1, \ldots O_t, q_t = S_i \mid \lambda)$ represents the probability of the partial observation sequence, $O_1, O_2, \ldots O_t$, and state S_i at time t, given the model λ. This definition is used to describe the health status of system. The calculation of $\alpha_t(i)$ is given in (9).

4.2. System Framework. The system is established based on three parts as shown in Figure 2. The first part trains the MG-HMM and FS-LSSVR model. According to the known samples, several MG-HMM models can be trained based on expectation maximization algorithm. Then, the forward variables of each sample can be calculated based on these MG-HMM models. The corresponding FS-LSSVR model is established with these forward variables. The MG-HMM models and corresponding FS-LSSVR models are constructed into a model library. The second part of the system is to recognize the unknown sample based on the model library. This part determines the MG-HMM model and FS-LSSVR with the maximization likelihood.

The third part of the system calculates the forward variables based on the MG-HMM obtained from the second part. These forward variables are put into the corresponding FS-LSSVR model to compute the RUL of the unknown sample.

5. Experiments

5.1. RUL Evaluations Metrics. Fault prognostic has its inherent particularity contrary to fault diagnosis. Therefore, Saxena et al. present some new evaluation metrics on fault prognostics [24–27]. After making a thorough inspection of these metrics, this paper further proposes two metrics for fault prognostics metrics: MAα and α-Nmap. We consider the series of evaluation metrics that could effectively measure algorithm performance in real-time RUL assessment.

First of all, a variable $r^f(t)$ is defined as the actual RUL when the system is in t moment, while variable $r^p(t)$ is defined as the prediction RUL at the same time. Hence, the prediction error percentage is represented as

$$\mathrm{PE}(t) = \frac{\left|r^p(t) - r^f(t)\right|}{r^f(t)}. \tag{28}$$

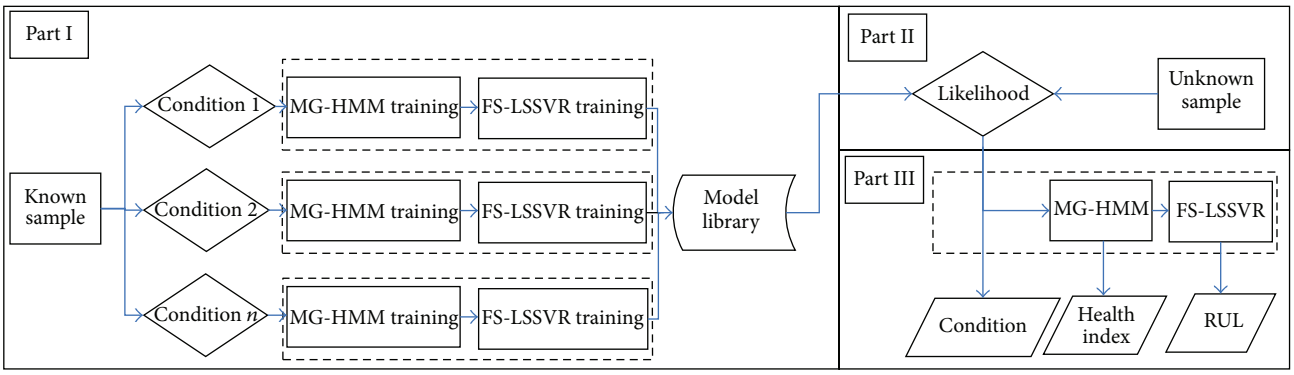


Figure 2: System framework.

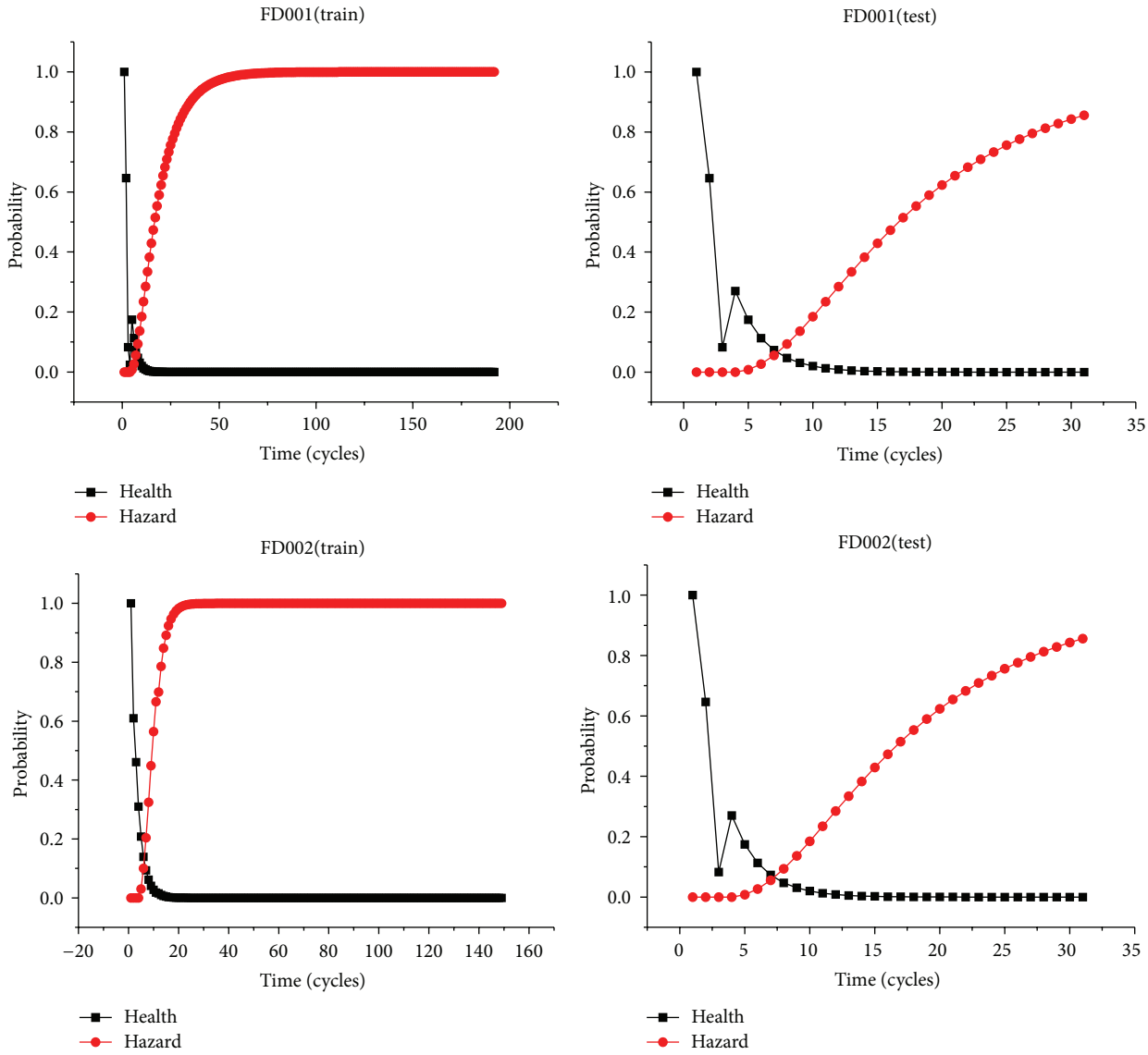


Figure 3: Continued.

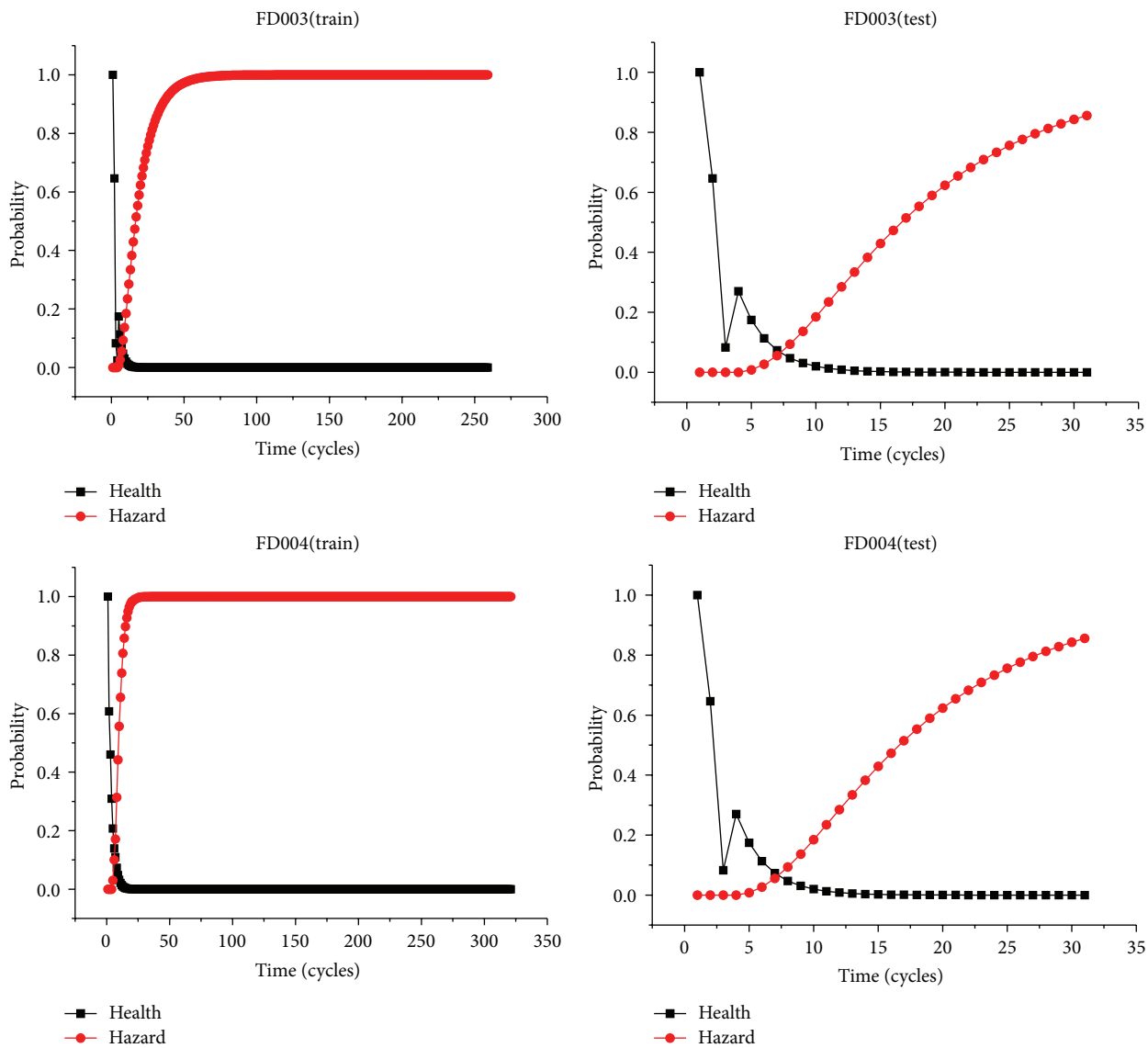


Figure 3: Health indexes for several samples.

Obviously there is PE ≥ 0. When the PE is equal to zero, there will be a perfect prediction.

(I) MAPE. Assumes the prediction time start that from $t = 1$ to the final failure moment $t = N$. Based on the prediction error percentage metrics, there are several evaluation metrics listed below

$$\mathrm{MAPE} = \frac{1}{N}\sum_{t=1}^{N}\mathrm{PE}(t). \tag{29}$$

The metrics measure the average prediction precision along with the time axis.

(II) BIAS. Consider the following:

$$\mathrm{BIAS} = \sqrt{\frac{1}{N-1}\sum_{t=1}^{N}(\mathrm{PE}(t) - \mathrm{MAPE})}. \tag{30}$$

The metrics measure the prediction bias along with the time axis.

(III) $\alpha(t)$, $\alpha \in [0,1]$. If there is $[1-\alpha]r^f(t) \leq r^p(t) \leq [1-\alpha]r^f(t)$, there will be $\alpha(t) = \text{true}$, otherwise $\alpha(t) = \text{false}$. The metrics gives a Boolean determination whether the prediction time is in a certain earlier or later range away from actual time.

(IV) MAα. Consider the following:

$$\mathrm{MA}\alpha = \frac{1}{N}\left(\sum_{t=1}^{N} I(\alpha(t))\right), \quad \alpha \in [0,1]; \tag{31}$$

where there is $\alpha(t) = \text{true}$ there will be $I(\alpha(t) = \text{true}) = 1$, otherwise, $I(\alpha(t) = \text{true}) = 0$. In fact, these metrics makes a simple statistics of $\alpha(t)$. The great advantage of the metrics

Table 1: C-MAPSS data set.

Data Set	Train trajectories	Test trajectories	Conditions	Fault modes
FD001	100	100	One (sea level)	One (HPC Degradation)
FD002	260	259	Six	One (HPC degradation)
FD003	100	100	One	Two (HPC degradation, fan degradation)
FD004	248	249	Six	Two (HPC degradation, fan degradation)

is that they do not need to consider specific prediction time point.

(V) α-Nmap. Furthermore, we specify $\lambda = 1/k$ and $\alpha = \lambda m$ where there is $m = \{1, 2 \ldots, k\}$. Hence, the α-Nmap is represented as

$$\alpha\text{-Nmap} = \frac{1}{k}\sum_{m=1}^{k} \text{MA}\alpha\,(m\lambda)\,. \tag{32}$$

The metrics mean the whole prediction correct percentage when the allowable prediction range changes in the interval $[0, 1]$. Therefore, there is α-Nmap $\in [0, 1]$. Furthermore, the best solution is obtained when α-Nmap $= 1$.

(VI) C. The metrics indicate the convergence performance of prediction. The detailed description can be provided as follows.

$$x_c = \frac{(1/2)\sum_{t=1}^{N-1}\left((t+1)^2 - t^2\right)\text{PE}\,(t)}{\sum_{t=1}^{N}\text{PE}\,(t)},$$
$$y_c = \frac{(1/2)\sum_{t=1}^{N-1}\text{PE}(t)^2}{\sum_{t=1}^{N}\text{PE}\,(t)}, \tag{33}$$
$$C = \sqrt{x_c^2 + y_c^2}.$$

5.2. Data Set. (C-MAPSS) Commercial modular aeropropulsion system simulation, [24] is a tool for simulating a realistic large commercial turbofan engine. The software simulates an engine model of the 90000 lb thrust class and an atmospheric model capable of simulating operations at (i) altitudes ranging from sea level to 40000 ft, (ii) March numbers from 0 to 0.90, and (iii) sea-level temperatures from −60 to 103°F. Several dataset generated by C-MAPSS has been uploaded on NASA website. The description of the dataset is shown in Table 1.

There are 26 columns in these data sets. Column one represents the unit number. Column two is the time of unit running. Columns 3–5 are the different unit operation conditions. The remaining columns are the 21 sensor measurements. The software calculates the health index according to four parameters of the 21 variables. But the researchers could not find these parameters explicitly.

The train trajectories contain all sensor measurements of one unit during its whole life. Otherwise, the test trajectories only provide part of all measurements of one unit. That is, the dataset are truncated at an appropriate time before its failure. The data set is divided into four training sets and four test sets. In the experiments, we used FD001 and FD002 train trajectories as training sets and FD001 and FD002 test trajectories as test set.

In order to train the MG-HMM model on data, we firstly transform the 24 sensor measurements (including 3 columns working conditions and 21 columns other sensor measurements) into values in $(0, 1)$ range by Sigmoid function $x = 1/(1 + \exp(x))$. For the considerations of performance and precision, here we choose the HMM model with five states and two Gaussian mixture models. The model training, unknown sample recognition, and RUL prediction process are listed as follows.

(1) Four individual MG-HMM models are trained on the four data sets with different work conditions.

(2) Health index of different data sets is calculated based on the corresponding MG-HMM.

(3) FS-LSSVR models corresponding to the MG-HMM are built based on health index, running cycle t, and RUL of training data set.

(4) Online recognition for unknown sample is done based on MG-HMM. In other words, the algorithm determines the unknown sample as the emission of the MG-HMM which has the maximization likelihood on the unknown sample.

(5) Health index of the time t is calculated based on the MG-HMM obtained in (4).

(6) Based on the time cycle t and health index, the FS-LSSVR corresponding to the MG-HMM obtained in (4) is used to predict the RUL.

5.3. Condition Recognition. These MG-HMM models are used to judge the work condition of unknown sample. The judgment results are listed in the first row of Table 2. From Table 2, we find the recognition accuracy on FD001 and FD002 is very high, while the recognition accuracy of FD003 and FD002 is very low. The reason is that the FD001 and FD002 only contain one type fault, that is, HPC degradation. However, the FD003 and FD004 contain two type faults; one is HPC degradation and the other is fan degradation. Furthermore, we give a deeper inspector for the dataset. Then we find that FD003 operates under the same condition as FD001 and FD004 operates under the same condition as FD002. Then we adjust the algorithm with respect to the work condition. The algorithm is considered to do right recognition if it judges the FD003 as the emission of MG-HMM1 which is trained from FD001. Similarly, the algorithm is thought to do right recognition if it judges the FD004 as

Table 2: Accuracy of condition recognition (%).

	FD001	FD002	FD003	FD004
Before adjustment	99.06	77.69	0.72	22.29
After adjustment	—	—	100.00	99.87

the emission of MG-HMM2 which is trained from FD002. After this adjustment, the accuracy of condition recognition for FD003 is improved from 0.72% to 100%. Meanwhile, the accuracy of condition recognition for FD004 is improved from 22.29% to 99.87%.

5.4. Health Indexes. We apply the various MG-HMM models obtained from the training data set into health index calculation on the unit 1 of training data set and testing data set. The health indexes are illustrated in Figure 3. In Figure 3, the health state is the first state of MG-HMM and the hazard state is the final state. We find the probability of health decreases as time goes on. Meanwhile, the probability of hazard increases as time goes on. The situation is in line with the actual working condition of the traditional equipment.

5.5. RUL Assessment. Once health indexes of unknown samples are calculated based on MG-HMM models, the RUL can be predicated based on FS-LSSVR models. Here we also make a comparison between ANN and FS-LSSVR. The reason we choose ANN is that ANN is a good nonlinear regression tool in machine learning area. The comparison results are listed in Table 3. From MAPE metric, we find that the FS-LSSVR can make a higher accuracy RUL prediction than ANN. Based on BIAS metric, we find the deviation of RUL prediction based on FS-LSSVR is lower than ANN. From the comparison result of MAα and α-Nmap, we find the FS-LSSVR can make more near RUL prediction than ANN. Also the FS-LSSVR converges to the failure point faster than ANN.

Table 3: RUL assessment comparison between ANN and FS-LSSVR.

Regression	Dataset	MAPE	BIAS	MAα	α-Nmap	C
ANN	FD001	0.43	0.43	0.29	0.07	6745.52
	FD002	0.48	0.70	0.29	0.07	8415.49
	FD003	0.44	0.36	0.24	0.07	10753.88
	FD004	0.49	0.53	0.23	0.07	11270.31
FS-LSSVR	FD001	0.38	0.33	0.37	0.07	5779.59
	FD002	0.44	0.55	0.33	0.07	7414.83
	FD003	0.40	0.27	0.30	0.07	10322.90
	FD004	0.47	0.40	0.27	0.07	11066.95

6. Conclusions

In this paper we present a hybrid method of MG-HMM and the FS-LSSVR for fault prognostic. The system contains three parts. The first part is responsible for training the MG-HMM models and FS-LSSVR models. The second part of the system recognizes the unknown sample based on the model library established in the first part. The third part of the system calculates the forward variables based on the MG-HMM obtained from the second part. These forward variables are inputted into the corresponding FS-LSSVR model to compute the RUL of the unknown sample.

Furthermore, after making a thorough inspection for former metrics of RUL assessment, this paper proposes two novel metrics for fault prognostics metrics: MAα and α-Nmap. From the comparison result of MAα and α-Nmap, we find the FS-LSSVR can make more near RUL prediction than ANN.

One interesting direction for future work is to investigate other state judgment techniques. The techniques can be developed based on pattern recognition techniques. Among these techniques the methods which can generate probability of health type are more preferred.

Another interesting direction for future work is to explore different regression methods. For example, the deep neural network (DNN) [25] and extreme learning machine (ELM) [28] are two newly developing machine learning algorithms. Maybe there is a better RUL prediction accuracy if the system framework can replace the FS-LSSVR with DNN or ELM. However, the calculation performance for large scale application of the two algorithms is still open issue, as we have considered in this paper.

Acknowledgments

The authors wish to thank the anonymous reviewers and the editors for their constructive suggestions. This paper is supported by the following funding: National High Technology Research and Development Program of China (863 Program) (2012AA062103); Jiangsu Province Innovation Fund prospective Joint Research Project (BY2012081); Jiangsu Province Science and Technology Achievement Transformation Project (BA2010058); Jiangsu Normal University key foundation (10XLA13).

References

[1] A. K. S. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mechanical Systems and Signal Processing*, vol. 20, no. 7, pp. 1483–1510, 2006.

[2] A. Heng, S. Zhang, A. C. C. Tan, and J. Mathew, "Rotating machinery prognostics: state of the art, challenges and opportunities," *Mechanical Systems and Signal Processing*, vol. 23, no. 3, pp. 724–739, 2009.

[3] M. Pecht, J. Mathew, L. Ma, A. Tan, M. Weijnen, and J. Lee, *A Prognostics and Health Management for Information and Electronics-Rich Systems Engineering Asset Management and Infrastructure Sustainability*, Springer, London, UK, 2012.

[4] X.-S. Si, W. Wang, C.-H. Hu, and D.-H. Zhou, "Remaining useful life estimation—a review on the statistical data driven approaches," *European Journal of Operational Research*, vol. 213, no. 1, pp. 1–14, 2011.

[5] J. L. Godwin, P. Matthews, and C. Watson, "Classification and detection of electrical control system faults through SCADA data analysis," *Chemical Engineering Transactions*, vol. 33, pp. 985–990, 2013.

[6] B. W. Lo, R. L. Macdonald, A. Baker, and M. A. Levine, "Clinical outcome prediction in aneurysmal subarachnoid hemorrhage using Bayesian neural networks with fuzzy logic inferences," *Computational and Mathematical Methods in Medicine*, vol. 2013, Article ID 904860, 10 pages, 2013.

[7] Z. Zhang, Y. Wang, and K. Wang, "Intelligent fault diagnosis and prognosis approach for rotating machinery integrating wavelet transform, principal component analysis, and artificial neural networks," *The International Journal of Advanced Manufacturing Technology*, vol. 68, no. 1–4, pp. 763–773, 2013.

[8] X. Zhou, Z. Xiang, M. Liu, and J. Xiang, "A multiwavelet support vector machine prediction algorithm for avionics PHM," in *Intelligent Computing Theories*, pp. 295–304, Springer, 2013.

[9] D. N. Thatoi, H. C. Das, and D. R. Parhi, "Review of techniques for fault diagnosis in damaged structure and engineering system," *Advances in Mechanical Engineering*, vol. 2012, Article ID 327569, 11 pages, 2012.

[10] C. Bunks, D. McCarthy, and T. Al-Ani, "Condition-based maintenance of machines using hidden Markov models," *Mechanical Systems and Signal Processing*, vol. 14, no. 4, pp. 597–612, 2000.

[11] P. Baruah and R. B. Chinnam, "HMMs for diagnostics and prognostics in machining processes," *International Journal of Production Research*, vol. 43, no. 6, pp. 1275–1293, 2005.

[12] R. B. Chinnam and P. Baruah, "Autonomous diagnostics and prognostics through competitive learning driven HMM-based clustering," in *Proceedings of the International Joint Conference on Neural Networks*, vol. 4, pp. 2466–2471, July 2003.

[13] F. Camci and R. B. Chinnam, "Health-state estimation and prognostics in machining processes," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 3, pp. 581–597, 2010.

[14] H. Ocak, K. A. Loparo, and F. M. Discenzo, "Online tracking of bearing wear using wavelet packet decomposition and probabilistic modeling: a method for bearing prognostics," *Journal of Sound and Vibration*, vol. 302, no. 4-5, pp. 951–961, 2007.

[15] A. H. Tai, W.-K. Ching, and L. Y. Chan, "Detection of machine failure: hidden Markov Model approach," *Computers and Industrial Engineering*, vol. 57, no. 2, pp. 608–619, 2009.

[16] Z.-J. Zhou, C.-H. Hu, D.-L. Xu, M.-Y. Chen, and D.-H. Zhou, "A model for real-time failure prognosis based on hidden Markov model and belief rule base," *European Journal of Operational Research*, vol. 207, no. 1, pp. 269–283, 2010.

[17] D. A. Tobon-Mejia, K. Medjaher, N. Zerhouni, and G. Tripot, "A data-driven failure prognostics method based on mixture of Gaussians hidden Markov models," *IEEE Transactions on Reliability*, vol. 61, no. 2, pp. 491–503, 2012.

[18] Z. Liu, Q. Li, X. Liu, and C. Mu, "A hybrid LSSVR/HMM-based prognostic approach," *Sensors*, vol. 13, no. 5, pp. 5542–5560, 2013.

[19] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[20] W. Li, Z. W. Wang, Z. C. Zhu, G. B. Zhou, and G. A. Chen, "Design of online monitoring and fault diagnosis system for belt conveyors based on wavelet packet decomposition and support Vector machine," *Advances in Mechanical Engineering*, vol. 2013, Article ID 797183, 10 pages, 2013.

[21] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.

[22] M. Espinoza, J. A. K. Suykens, and B. De Moor, "Fixed-size least squares support vector machines: a large scale application in electrical load forecasting," *Computational Management Science*, vol. 3, no. 2, pp. 113–129, 2006.

[23] C. Williams and M. Seeger, "Using the nystroem method to speed up Kernel machines," *Advances in Neural Information Processing Systems*, vol. 13, pp. 682–688, 2001.

[24] A. Saxena and K. Goebel, "C-MAPSS Data Set," NASA Ames Prognostics Data Repository, NASA Ames, Moffett Field, Calif, USA, 2008, http://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/.

[25] R. Collobert and J. Weston, "A unified architecture for natural language processing: deep neural networks with multitask learning," in *Proceedings of the 25th International Conference on Machine Learning*, pp. 160–167, July 2008.

[26] A. Saxena, J. Celaya, B. Saha, S. Saha, and K. Goebel, "Metrics for offline evaluation of prognostic performance," *International Journal of Prognostics and Health Management*, vol. 1, no. 1, p. 20, 2010.

[27] A. Saxena, J. Celaya, E. Balaban et al., "Metrics for evaluating performance of prognostic techniques," in *Proceedings of the International Conference on Prognostics and Health Management*, pp. 1–17, 2008.

[28] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 607086, 9 pages
http://dx.doi.org/10.1155/2013/607086

Research Article

Automatic Detection of Repetitive Components in 3D Mechanical Engineering Models

Laixiang Wen,[1] Jinyuan Jia,[2] Shuang Liang,[2] and Jianhua Zhang[3]

[1] *School of Electronics and Information Engineering, Tongji University, Shanghai 201804, China*
[2] *School of Software Engineering, Tongji University, Shanghai 201804, China*
[3] *College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China*

Correspondence should be addressed to Shuang Liang; shuangliang@tongji.edu.cn

Received 15 August 2013; Accepted 23 September 2013

Academic Editor: Shengyong Chen

We present an intelligent method to automatically detect repetitive components in 3D mechanical engineering models. In our work, a new Voxel-based Shape Descriptor (VSD) is proposed for effective matching, based on which a similarity function is defined. It uses the voxels intersecting with 3D outline of mechanical components as the feature descriptor. Because each mechanical component may have different poses, the alignment before the matching is needed. For the alignment, we adopt the genetic algorithm to search for optimal solution where the maximum global similarity is the objective. Two components are the same if the maximum global similarity is over a certain threshold. Note that the voxelization of component during feature extraction and the genetic algorithm for searching maximum global similarity are entirely implemented on GPU; the efficiency is improved significantly than with CPU. Experimental results show that our method is more effective and efficient than that existing methods for repetitive components detection.

1. Introduction

Pattern recognition plays an important role in artificial intelligence. In mechanical engineering, there are always repetitive patterns (components) in designs. The ability to detect such repetitive components automatically from the general input of 3D mechanical models is important for many applications in mechanical engineering. For example, finding out such repetitive components provides a natural way of model reduction [1, 2], and it facilitates intelligent editing of mechanical models, such as propagating edits to multiple instances of one mechanical component.

Our paper addresses the problem of detect repetitive components in 3D mechanical models, such as the wheel rims and the tires of a car, shown in Figure 1. This means we are looking for components in the model that can be reused under orthogonal mappings (rotation and mirroring) and translations to assemble the original model more compactly. Detecting such repetitive components in 3D models is a challenging work, since we typically have no prior knowledge of the size, shape, location, or orientation of the individual components. It is essential to normalize the pose of each component into a canonical coordinate system before the similarity measuring between each other. The pose alignment [3–5] is a series of geometry-invariant transformations, including translation, rotation, and scaling. After the pose alignment, the feature descriptors [5–11] are extracted to represent each component. The distance between the feature descriptors indicates the similarity between the two components.

Content-based 3D model retrieval [8, 12–15] also faces the same problem of pose alignment and similarity measuring. The most well-known approach to compute the pose alignment in 3D model retrieval is the Principal Component Analysis (PCA) method [9, 16], which is based on the computation of moments of the components. In most cases, 3D models are represented by geometric meshes and corresponding attributes which are used as input of PCA, such as vertices (CPCA) [1, 5], normals (NPCA) [5], and voxels (VPCA) [11]. However, PCA often faces severe inaccuracies regarding the alignment of similar models [10, 14]. Similarity measuring can be separated from pose alignment if it is based on PCA

method. Feature descriptor used for similarity measuring differs from one method to another. Chen et al. [17] proposed the light field descriptor with good discriminative power but it has long extraction and matching time as a disadvantage, generated by 2D images-projections of the 3D model. Zhang et al. [11] use hybrid descriptors being extracted from these depth images to represent the 3D model. These feature descriptors usually are based on pose alignment which could be separated from similarity measuring. There are also some optimization methods [18, 19] which mix the pose alignment and the similarity measuring. They are dedicated to finding an optimal solution of the pose alignment under which the maximum similarity can be achieved. The pose is adjusted through the resulting similarity. These optimization methods outputs better results than PCA-based methods, but have higher time cost. To minimize the time cost, Martinek and Grosso [20] propose a GPU-based similarity measuring function using improved depth images to accelerate optimal pose searching.

The inaccuracy of PCA-based pose alignment affects severely the similarity measuring, while the time efficiency is the problem for optimization methods but it has better accuracy. Our aim is to detecting repetitive components in 3D mechanical engineering models, so that the accuracy is the highest priority with the consideration of time efficiency. In our work, a new Voxel-based Shape Descriptor (VSD) is proposed for the effective matching, based on which a similarity function is defined. It uses the voxels intersecting with 3D outline of mechanical components as the feature descriptor. Because each mechanical component may have different pose, the alignment before the matching is needed. For the alignment, we adopt the Genetic Algorithm (GA) to search the optimal solution where the maximum global similarity is the objective. If the maximum global similarity is over 0.99 allowing the error 0.01, we tell that the two components are same. Note that the voxelization of component during feature extraction and the GA for searching maximum global similarity are entirely implemented on GPU; the efficiency is improved significantly than on CPU. With GPU, the time efficiency of our method is very close to that of nonoptimization methods.

2. Related Work

2.1. Pose Alignment. Pose alignment of 3D models is used for preprocessing models before 3D shape matching and is commonly used in 3D model retrieval [8, 13, 21]. To conduct such alignment, a concatenation of isometrics in 3D space must be selected to determine the canonical coordinate system, and then the related model is aligned under it. The most well-known approach used to compute the canonical coordinate system of 3D models is the Principal Component Analysis method (PCA) [9], which is based on the computation of moments of 3D models.

After translating the 3D model according to its center of mass to the origin of the world coordinate system, three principal axes computed with PCA are used to determine the orientation. Some PCA-based methods, such as CPCA [9, 12] and NPCA [5], mainly focus on the different selection

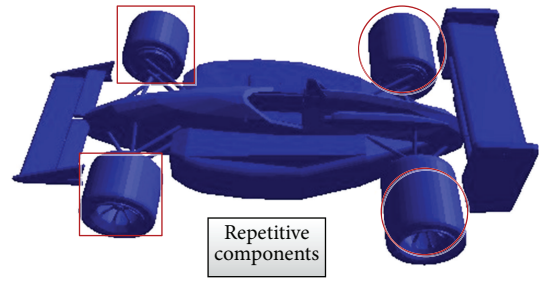


Figure 1: The aim of our work to finding out repetitive mechanical components, for example, tires and wheel rims.

of samples; the samples of CPCA are the weighted vertices, whose precision is affected by the distribution of the vertex on the surface. While NPCA uses the area-weighted normal of each triangle as the sample, its precision depends on the distribution of the normal. Since PCA is a statistic method, its precision, regarding to the alignment of similar models with small difference [10, 14], is affected by the quality of the samples.

There are also other different solutions. First, symmetry-based approach, Chaouch and Verroust-Blondet [4], is based on two types of symmetries of the models: the reflective symmetry [22] and the local translational symmetry along a direction. Second, optimization-based approach, Kazhdan [18] decomposes the 3D optimization alignment problem into two subproblems of 2D and 1D optimization aligning respectively. Martinek and Grosso [20] introduced a GPU-based optimal 3D model alignment approach. It is similar to Kzahdan's method, but a GPU-based similarity function using improved depth images to accelerate approximately optimal pose alignment searching and high efficiency is obtained. Third, projection area-based approach, Johan et al. [3] proposed an algorithm to find the alignment axes based on minimum projection area by noting that many objects have a minimum projection area when we orthogonally project them in the direction parallel to one of the axes of the ideal canonical coordinate frame.

2.2. Feature Descriptor. Feature descriptor is extremely important for model matching, model classification, model indexing, or model recognition. Because 3D space can be parameterized using spherical functions, Vranic and Saupe [12] proposed the ray-based descriptor which characterizes a 3D model by a spherical extent function capturing the furthest intersection points of the model's surface with rays emanating from the origin. Spherical harmonics and moments representation of the spherical extent function were tested in [6], showing better performance than the former. Chen et al. [17] proposed the light field descriptor with good discriminative power but it has long extraction and matching time as a disadvantage, generated by 2D images-projections of the 3D model. In [23], the generalized radon transform is introduced, where the descriptor is extracted by combining the radial integration transform which captures the radial information of a 3D model and the spherical integration transform which captures its spherical information. Cornea

et al. [7] proposes the use of curve skeletons produced by application of the generalized distance field to the volumetric representation of a 3D model. The hybrid descriptor proposed by [5] is a combination of many different methods in order to achieve better performance. Zhang et al. [11] use hybrid descriptors being extracted from these depth images to represent the origin 3D model shape. More other methods were surveyed in [13].

3. Overview of the Proposed Approach

We proposed a high-efficient and accurate approach to detect the repetitive components. Matching method for 3D geometries is our main problem to be solved. Voxel-based Shape Descriptor (VSD) which represented each component is used for our similarity measuring. Pose alignment is the preprocessing for the similarity measuring. Our approach uses Genetic Algorithm (GA) to achieve the maximum similarity by searching optimal pose of two components. But the time cost is the fatal weakness of using genetic algorithms directly on CPU. GPU has the highly parallel architecture which can process a large amount of data in the same time cycle of CPU, so we transplant our GA from CPU to GPU to reduce the runtime. The projected depth images are usually used as the feature descriptor, and it is also used in Martinek's method [20]. But projecting these depth images must use the GPU rendering pipeline, so it will affect the whole algorithms if it was run in GPU as general purpose. The VSD we proposed could apply the spatial transformation quickly and easily running on GPU.

Our matching method is transformed into the problem of using GA to search optimal pose alignment of two components to achieve maximum similarity, shown in Figure 2. For two components A and B, do the following.

(1) Simply normalize the A and B to their norm-size by their Axis Aligned Bounding Box (AABB), so the domain of the translation and scale are limited into a subdomain of 3D space. It minimizes the domain of GA searching, shown in Figure 2(a).

(2) Do the voxelization of A and B according to their own AABB, so that the voxels intersecting the 3D outline of A and B can be used for our proposed VSD, shown in Figure 2(b). It stands for the physic shape of component, so it works when detecting rigid repetitive components. And the significant feature is that it can be paralleled on GPU. Since the resolution of voxelization affects the accuracy of similarity measuring, the discussion of it is detailed in Section 6.1.

(3) Initialize the population of poses of B for GA method running on GPU, since we align B to A, shown in Figure 2(c). Better initial population for GA method can reduce the iteration of generation to find the optimal solution. So the PCA computed poses such as CPCA and NPCA, special rotation angles such as 0.5π, π, and 2π around axes x, y, and z, and the original pose of B are used as initial population, more details in Section 5.2.

(4) Search the optimal pose alignment of A and B to achieve the maximum similarity using GPU-based GA method, and update the evolution poses, shown in Figure 2 (central part). Limiting the searching domain is very significant to time efficiency and accuracy of the GA method. Section 4.1 describes the definition of the domain.

(5) Get the optimal pose alignment of A and B, shown in Figure 2(d).

4. Similarity Measuring

4.1. Pose Definition. In order to find the best pose alignment between A and B, we let A reside in its original pose and assign a pose P only to B. First of all, both of the two components are normalized by their AABB bounding box. The quality of the pose alignment can be measured by means of the similarity function using VSD defined in the next section which we will also be refered to as objective function.

The pose in 3D space can be composed of three parts: rotation, translation, and scale. Translation uses the center position of the 3D component as the reference point, $\mathbf{t} = \{t_x, t_y, t_z\}$. Scale affects the size of B, and it can be represented by a scalar to complete the scaling, $\mathbf{s} = \{s\}$. Notice that both of the two components are normalized, so the searching domain of the translation and scale, respectively, are confined into a subdomain of the 3D space. For the translation, if B is moved out of the range of the AABB of A, the two components will never overlap. So we can limit the translation vector $\mathbf{t}$ by the domain $x, y, z \in [-1.0, 1.0]$. The same to the scale, if B is too small or too large, the alignment result will obviously be the solution, so the domain of the scale can be confined to $s \in [0.5, 2.0]$. For the rotation, it can be represented by an orthogonal 3×3 matrix which is not suitable for carrying out numerical optimization procedure. We parameterize the space of rotations according to Euler angles $\theta \in [-0.5\pi, 0.5\pi]$ and $\phi, \varphi \in [-\pi, \pi]$, which are similar to the ways used in [18]. It expresses every rotation around the y-axis, multiplied on the left and right by rotation about the z-axis, $\mathbf{r}(\theta, \phi, \varphi)$. Now we obtain the pose definition of the component in 3D space by P:

$$P = \{\mathbf{r}, \mathbf{t}, \mathbf{s}\} = \begin{bmatrix} \cos\phi\cos\varphi + \sin\phi\sin\theta\sin\varphi & \sin\varphi\cos\theta & -\sin\phi\cos\varphi + \cos\phi\sin\theta\sin\varphi & 0 \\ -\cos\phi\sin\varphi + \sin\phi\sin\theta\cos\varphi & \cos\varphi\cos\theta & \sin\varphi\sin\phi + \cos\phi\sin\theta\cos\varphi & 0 \\ \sin\phi\cos\theta & -\sin\theta & \cos\phi\cos\theta & 0 \\ t_x & t_y & t_z & s \end{bmatrix}. \quad (1)$$

The goal of our approach is to find the pose alignment P which applied to B could get the maximum similarity.

4.2. Voxel-Based Shape Descriptor. Voxelization is a widely used technique to adjust the state of optimal solution of similarity using GA and capture the main feature of the 3D model. Many 3D model retrieval approaches [24, 25] use it as a basic way to extract the feature descriptor. For our algorithm, we propose a similarity function using VSD for adjusting population of poses, which is highly accurate and efficient due to the fact that we implement the method entirely on the GPU.

Initially, the smallest AABB circumscribing the component is constructed, and then both the model and this AABB are normalized into unit cube. Decompose this AABB into $N \times N \times N$ uniform grid of voxels and set each voxel with the initial state as transparent. Here, we use $\text{voxel}(x, y, z, s)$ to describe the state of one voxel, where (x, y, z) is the spatial position of this voxel and s is the scale size of it. If the initial voxel located at (x, y, z) is regarded as an opaque voxel, if there are geometric primitives intersecting this voxel, there is a voxel $(x, y, z, 1) = 1$. Otherwise, it is regarded as transparent, $\text{voxel}(x, y, z, 1) = 0$. All the opaque voxels will be gathered to form a union to describe the shape of original 3D component and be used to perform the similarity measuring:

$$\mathbf{V} = \{\text{voxel}(x, y, z, s) \mid \text{voxel}(x, y, z, s) = 1\}, \tag{2}$$

where $1 \leq x, y, z \leq N$. We denote the voxels collections of A and B as the sets A_V and B_V, respectively. We define a $\text{voxel}(x, y, z)$ to belong to the union of A_V and B_V ($A_V \cap B_V$), only if this voxel that belonged to A_V intersects with at least one voxel in B_V. Then the similarity measuring can be defined as

$$\text{similarity}(A, B) = \frac{|A_V \cap B_V|}{|A_V \cup B_V|}. \tag{3}$$

For each potential pose of B, the spatial position must be applied to B and update the variables of similarity function. Our voxel-based similarity function does not depend on the rendering pipeline and could apply the pose to the voxels union quickly and independently. For the voxel $\text{voxel}(x, y, z) \in B_V$, if the current pose of B is P, we should apply this pose to the voxel as follows:

$$\begin{aligned}\text{voxel}' &= \text{voxel} \times P \\ &= [x, y, z, 1] \\ &\quad \times \begin{bmatrix} \cos\phi\cos\varphi + \sin\phi\sin\theta\sin\varphi & \sin\varphi\cos\theta & -\sin\phi\cos\varphi + \cos\phi\sin\theta\sin\varphi & 0 \\ -\cos\phi\sin\varphi + \sin\phi\sin\theta\cos\varphi & \cos\varphi\cos\theta & \sin\varphi\sin\phi + \cos\phi\sin\theta\cos\varphi & 0 \\ \sin\phi\cos\theta & -\sin\theta & \cos\phi\cos\theta & 0 \\ t_x & t_y & t_z & s \end{bmatrix},\end{aligned} \tag{4}$$

where voxel' is the new position of the voxel under the pose P and s is the size of the voxel. Since each voxel is a spatial axis-aligned cube with different size, the similarity computing become the intersection testing between these two voxels collections of the two objects. The whole voxels set of one object could also be divided into small parts to do the intersection testing independently, this makes our approach high parallelized and convenient to implement on GPU.

5. Optimal Solution of Similarity Using GA

The problem of finding the optimal pose alignment between two 3D components can now be formulated as a global optimization problem, where the task is to determine the pose P to achieve the maximum similarity value in the limited 3D space. Blinded searching in the potential solution domain has a very low efficiency, even if the solution domain is limited. One more efficient way to do the heuristic searching is the optimal algorithms, so here we use the genetic algorithm.

5.1. Our GA on GPU. Genetic algorithm is a powerful, domain independent search technique inspired by Darwinian Theory. In general, GA employs selection, mutation, and crossover to generate new search points in a state space. A genetic algorithm starts with a set of individuals that forms a population. Usually, the initial population is generated randomly using a uniform distribution or some heuristic strategy. On every iteration of the algorithm, each individual is evaluated using the fitness function and the termination function is invoked to determine whether the termination criteria have been satisfied. The algorithm ends if acceptable solutions have reached. Otherwise, the individuals in the population are manipulated by applying different evolutionary operators such as mutation and crossover. The consecutive process of replacement forms a new population for the next generation.

Although GA is very effective in solving many practical optimal problems, its execution time can become a limiting factor for some huge problems due to a lot of candidate solutions. Fortunately, the most time-consuming fitness evaluation can be performed independently for each individual in the population using various types of parallelization [26]. In our work, we implement the GPU-based GA to do the optimal aligning pose finding. The structure of our GA mapping onto the GPGPU architecture is shown in Figure 3.

In the structure on GPU shown in Figure 3, each potential pose of component B is the individual in the GA's population.

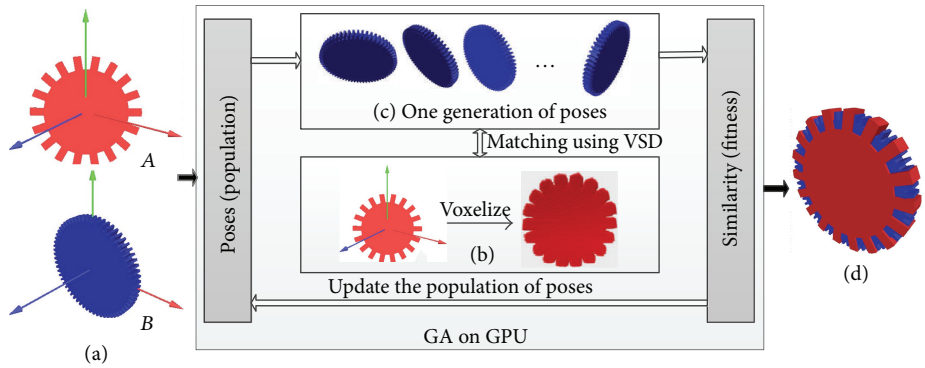


Figure 2: Searching optimal pose alignment to achieve maximum similarity.

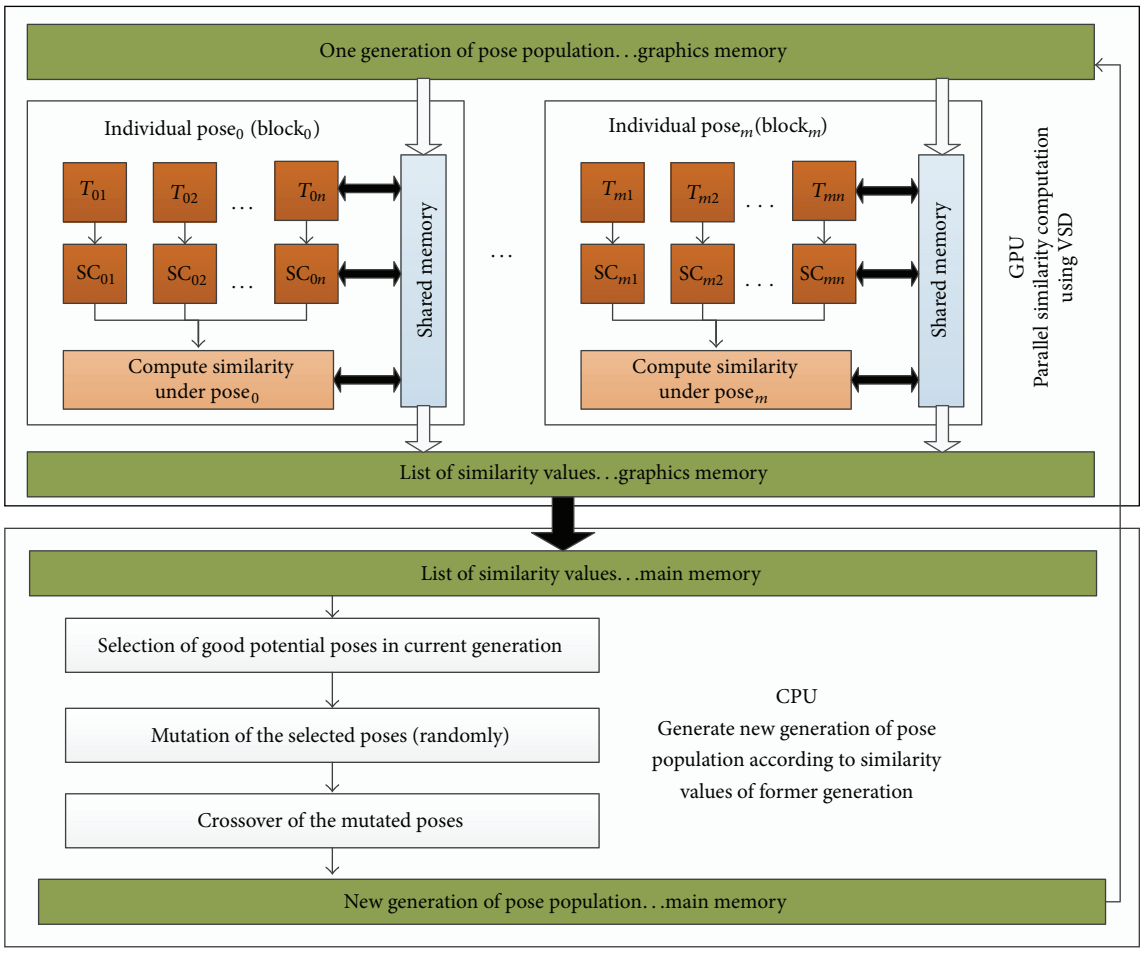


Figure 3: The parallel running structure of our GPU-based genetic algorithm.

For each block (block$_i$) in GPU, it computes the similarity under one individual pose in the population independently. In each block, the similarity is divided into many subsimilarity computations (SC_{ij}) mapping to each thread (T_{ij}) in the block. All the threads are synchronized to compute each final similarity under the corresponding individual pose. The work of CPU performs the evolution process of GA to generate new generation of pose population, including selection, mutation, and crossover, according to similarities of last generation. For our problem, if the any similarity is over 0.99, the process ends. Otherwise, the maximum count of iteration is reached.

5.2. Heuristic Initial Population. In general, if the generations of the GA are enough, it could find the optimal solution in theory. But in order to reduce the iterations of generation, we could use some heuristic strategies to generate the better

Table 1: Test data under different resolution of voxelization; size of population is 1024 and maximum generation is 150.

Resolution	CPU (sec) Average	GPU (sec) Average	Speedup (GPU/CPU) Average	Similarity (%) Average/worst
32	71.179	0.470	151.582	33.7/17.3
64	289.718	1.051	275.705	69.5/51.6
128	679.578	1.785	380.686	90.1/70.0
256	2652.107	4.741	432.824	90.1/70.0

initial population. If the initial population contains more solutions contributing potentially to the optimal solution, the GA searching will terminate more sooner. For the pose of B, there are some potential poses which may be the optimal pose:

(1) the original pose of component B: the original pose of the component may be in accordance with the perceptive system of the human, so in the pose the two components may have the largest union. Consider the following:

$$P = \{r(0,0,0), t(0,0,0), s(1)\}; \tag{5}$$

(2) the PCA pose of component B, transforming the pose of B from the original pose of B to the pose of A using PCA method: PCA is a quick and efficient way to approximately align the components into a canonical coordinate system. It is usually the potential solution of pose alignment to get the maximum similarity. Consider the following:

$$P = P^{A}_{\text{pca}} \cdot P^{B}_{\text{pca}}; \tag{6}$$

(3) the special rotation poses of component B, such as rotations around x, y, z axes with angles, 0.5π, π, 2π, and so forth: before the components are put into the database or used, many of their transformations are simply changed with the directions of the axes or special rotation angles. Hence, the poses to inverse these operations are gathered to form the thirdly initial poses. Consider the following:

$$P = \{r(\alpha, \beta, \gamma), \mathbf{t}, \mathbf{s}\}, \tag{7}$$

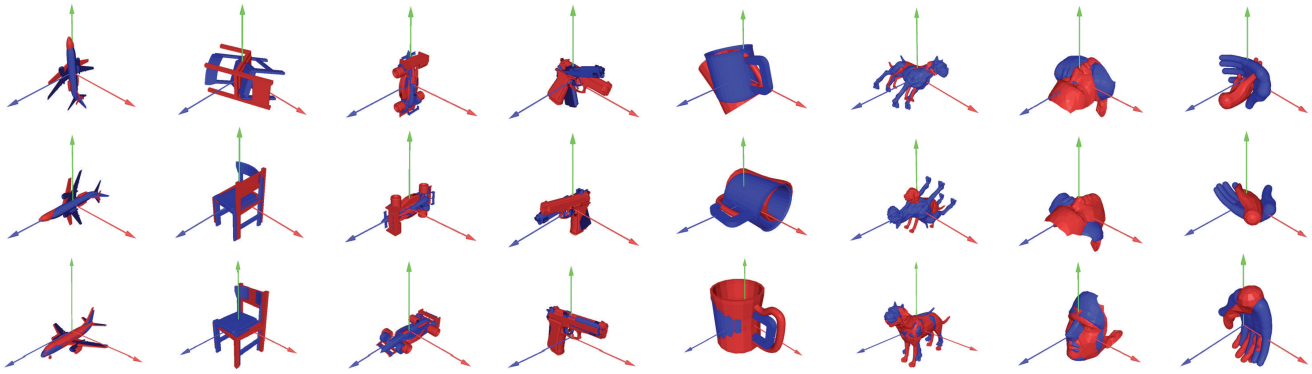
where $\alpha, \beta, \gamma \in \{\pi/6, \pi/4, \pi/3, \pi/2, \ldots, \pi\}$. These initial poses are combined randomly, inserted into the initial population of GA.

6. Experimental Results

To evaluate our proposed method, there are three aspects including VSD and its time efficiency, optimal pose alignment using GA method, and the ability to detect repetitive components in mechanical engineering models. The Princeton 3D Shape Benchmark database [27] is used for the first two tests, which is a model database dedicated to evaluate 3D model matching method. Note that one model in PSB stands for one component named in this paper. Models collected freely from the Internet are used for the third test that detect repetitive components. The hardware configuration we use is Intel Core 2 Duo 2.0 GHz, 1 G RAM, and Nvidia Quadro FX3800M.

6.1. Our VSD and Time Efficiency. Since we use the VSD for similarity computation, the resolution of voxelization affects the accuracy of representation of a component. So what resolution we should adopt needs to be determined. And the time efficiency is also an important aspect of our optimization method using GA for optimal pose alignment. Under different resolution of voxelization, the time cost of execution on CPU and GPU, speedup of transplanting from CPU to GPU, and similarity value for each matching between two components are recorded. We choose 100 models randomly from PSB, and the average results are shown in Table 1. From the results, we can see that the average and the worst of similarity value tend to be stable when the resolution of voxelization exceeds 128, so we use resolution $128 \times 128 \times 128$ in the following experiments. One hundred models chosen randomly from PSB were used to compute the similarity between each other under different voxel resolution of the VSD. The low resolution means the 3D outline of component is represented by the coarse voxels. Along with raising the resolution, the VSD describes the 3D outline more and more accurate. So each comparison could reach its maximum similarity value after the resolution of VSD exceeds some value. And the average similarity value and the worst of all comparisons' results tend to be stable. That is the way we determine that resolution value $128 \times 128 \times 128$ is sufficient to the VSD to describe the component accurately. From Table 1, the average time cost of execution based on GPU is fairly small to be an optimization method using GA. And we can also find that it gains a high speed-up of transplanting from CPU to GPU.

6.2. Evaluations on Our Optimal Pose Alignment. A comparison of our pose alignment is restricted to the most commonly used normalization technique, the PCA. Currently, there are two typical implementations of PCA, CPCA, and NPCA. The former uses the vertices of the 3D meshes to conduct the PCA. The latter uses the normal of each face or each vertex of 3D meshes to conduct the PCA. As to solving the PCA, the eigenvalues and eigenvectors of the covariance matrix of the samples, vertices (CPCA), or normals (NPCA) are needed to compute. Leting the eigenvalues be $\delta_1, \delta_2, \delta_3$, $\delta_1 \geq \delta_2 \geq \delta_3$, the corresponding eigenvectors to eigenvalues are $\mathbf{e_1}$, $\mathbf{e_2}$, $\mathbf{e_3}$. Because three principal axes are not oriented, firstly, these eigenvalues are sorted by descending order to make it



(a) Airplane (b) Chair (c) Car (d) Gun (e) Cup (f) Dog (g) Head (h) Hand

Figure 4: Alignment examples of PCA methods and ours; CPCA, NPCA, and ours are aligned from top to bottom.

Table 2: Resulting data on several typical classes of models in PSB.

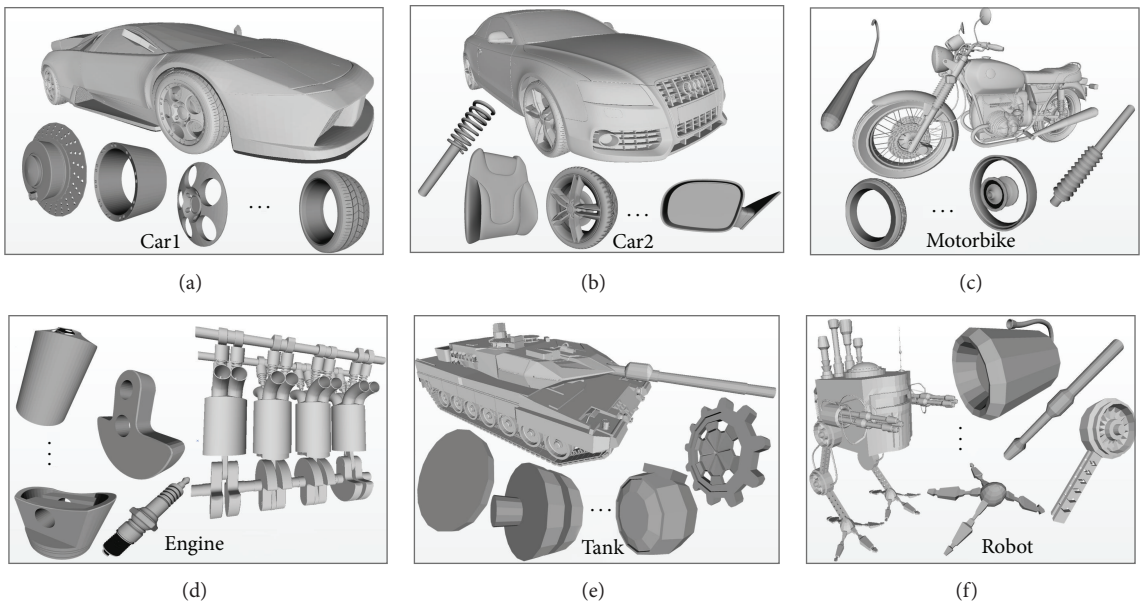
Classes of models	Model number	CPCA (%) Best/average/worst	NPCA (%) Best/average/worst	Our method (%) Best/average/worst
Vehicle	70	1.0/94.7/66.0	1.0/95.9/67.4	1.0/98.1/75.3
Airplane	99	1.0/71.7/11.9	1.0/89.6/54.3	1.0/97.5/90.1
Liquid container	24	1.0/65.4/37.1	1.0/66.8/34.7	1.0/98.2/80.5
Tables	35	1.0/80.2/13.4	1.0/81.8/14.7	1.0/94.3/87.5
Animals	124	1.0/91.5/55.7	1.0/89.6/54.3	1.0/95.1/70.2
Chairs	50	1.0/67.9/29.5	1.0/71.7/30.2	1.0/96.5/84.3
Body part	55	1.0/51.5/32.7	1.0/54.5/33.9	1.0/81.4/71.1

correspond to the x, y, and z axes, and then, we compute the similarity values for the rest of the six permutations of the axes' orientation. Among them, we take the orientation which has the maximum similarity value as the estimated PCA alignment. The major drawback of PCA is that the axes react sensibly to small changes in the geometry. This happens when the model has many small decoration parts. Furthermore, when the PCA confronts the degenerated cases, like sphere and cylinder, three principal axes are not certain [1, 2]. The visual results of some examples using different pose alignment methods are shown in Figure 4. Obviously, our method can get better pose alignment than the PCA methods. And our method can be applied to not only the man-made objects but also the natural objects.

The PCA results are shown in Table 2. It explains the unsatisfactory worst-case results and shows that our approach could provide a better alignment. In order to test our approach more objectively, let the population size and the generation size be large enough so that we can get the optimal solution of pose alignment to achieve maximum similarity in theory. The resolution of the voxelization is $128 \times 128 \times 128$, the size of pose population is 1024, and the max iteration count of generation is 150. We select several typical classes of models from PSB to test our method. The PSB offers some standard class files for the researchers to benchmark their methods. From the standard class file, we choose part of classes for our test and each class contains certain number of models determined by PSB. The average similarity and the worst case on each class of models are used to indicate the accuracy of pose alignment between two models. The best case of similarity means align with itself, so its value is always 1.0. The average similarity indicates the overall effect of the pose alignment method. The worst case significantly indicates if the pose alignment method has a strong ability. The result data is shown in Table 2. From the average similarity of each alignment method, our method is more stable under different cases than the other two methods. The worst case of each method shows that our method can always get an optimal pose alignment.

6.3. Ability to Detect Repetitive Components. Table 3 shows experimental results to measure our method and the method proposed by Papadakis et al. [5]. The models used in the experiments and the corresponding repetitive components searched out are shown in Figure 5.

To detect the repetitive components, the similarity measuring is of utmost importance. We propose a method based on GA to adjust the pose, so that the global maximum similarity is obtained. According to the count of the components before and after removing repetition using the two algorithms shown in Table 3, ours is more able to detect repetitive components of a model than Cai's. In other words, our method of similarity measuring is more accurate.



(a) (b) (c)

(d) (e) (f)

Figure 5: The models and the corresponding repetitive components searched out using our method.

Table 3: Result data of detecting repetitive components using our method and Cai's.

Models	Component count (original model)	Component count (ours) (removing repetition)	Component count (Cai's) (removing repetition)
Car1	360	145	236
Car2	407	208	300
Motorbike	487	236	307
Engine	189	50	121
Tank	1653	199	280
Robot	55	30	43

There are two main reasons that why our method is more accurate than Cai's.

(1) Our optimal pose alignment method has better efficacy than the PCA methods. The optimal pose alignment is obtained only if the two components have the global maximum similarity. It is not influenced by the resolution and the vertex distribution of the two components, while the PCA methods are sensitive to them.

(2) Our proposed feature descriptor VSD can be used to recognize the same components but different resolution, and can exactly represent the 3D outline of each component. While the vertex-to-vertex matching through nearest searching is used by Cai, it has a higher requirement of effect on pose alignment where the PCA methods usually fail when the two components are not exactly the same.

For the method for finding repetitive components, due to the elimination of redundancy, file size of model is obviously reduced and we can use lesser required components to restore the original model such that high compression ratio is obtained. It is worth noting that the compression ratio of a 3D model is determined by the number of repetitive components being found and whether these components contain a large scale of vertices and planes. Another application of this automatic detection of repetitive component may be the intelligent editing, and we believe that the potential applications of it are not confined to these.

7. Conclusions

In this paper, we introduced an automatic detection method of repetitive components in 3D mechanical engineering models but not limited to it. The global similarity between components is computed through a GA-based optimization method. It adjusts the pose to get an optimal pose to achieve the maximum global similarity. For similarity computation, a new VSD is proposed, using the 3D outline of components as the feature descriptor. To solve the time efficiency problem of the optimization method using GA, we implement on GPU for large data computation. And some heuristic strategies for initializing the population are also used to improve the quality of the results and to speedup the searching. The experimental results show that time efficiency, the effect of pose alignment, and the ability of repetitive components detection are all remarkably superior to existing methods. In the future, more accurate similarity computation will be researched, such as using the segmentation techniques to flag different parts of the 3D object with different weights. On the other hand, taking the advantage of GPU, more parallelized algorithms will be imported into the domain of mechanical engineering.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this article.

Acknowledgments

This research work was supported by the National Twelfth Five-year Plan Major Science and Technology Project of China (no. 2012BAC11B01-04-03). The authors also would like to thank the anonymous reviews for their valuable comments and suggestions.

References

[1] D. Shikhare, S. Bhakar, and S. Mudur, "Compression of large 3D engineering models using automatic discovery of repeating geometric features," *Signal Processing*, vol. 19, no. 20, p. 15, 2001.

[2] K. Cai, W. Wang, Z. Chen, Q. Chen, and J. Teng, "Exploiting repeated patterns for efficient compression of massive models," in *Proceedings of the 8th International Conference on Virtual Reality Continuum and its Applications in Industry (VRCAI '09)*, pp. 145–150, December 2009.

[3] H. Johan, B. Li, Y. Wei, and I. Iskandarsyah, "3D model alignment based on minimum projection area," *Visual Computer*, vol. 27, no. 6–8, pp. 565–574, 2011.

[4] M. Chaouch and A. Verroust-Blondet, "Alignment of 3D models," *Graphical Models*, vol. 71, no. 2, pp. 63–76, 2009.

[5] P. Papadakis, I. Pratikakis, T. Theoharis, G. Passalis, and S. Perantonis, "3D object retrieval using an efficient and compact hybrid shape descriptor," in *Proceedings of the Eurographics Workshop on 3D Object Retrieval*, 2008.

[6] D. Saupe and D. V. Vranić, "3D model retrieval with spherical harmonics and moments," in *Pattern Recognition*, pp. 392–397, Springer, New York, NY, USA, 2001.

[7] N. D. Cornea, M. Fatih Demirci, D. Silver, A. Shokoufandeh, S. J. Dickinson, and P. B. Kantor, "3D object retrieval using many-to-many matching of curve skeletons," in *Proceedings of the International Conference on Shape Modeling and Applications (SMI '05)*, pp. 368–373, June 2005.

[8] Y.-B. Yang, H. Lin, and Q. Zhu, "Content-based 3D model retrieval: a survey," *Chinese Journal of Computers*, vol. 27, no. 10, pp. 1297–1310, 2004.

[9] E. Paquet, M. Rioux, A. Murching, T. Naveen, and A. T. Ali Tabatabai, "Description of shape information for 2-D and 3-D objects," *Signal Processing*, vol. 16, no. 1, pp. 103–122, 2000.

[10] B. Bustos, D. A. Keim, D. Saupe, T. Schreck, and D. V. Vranić, "Feature-based similarity search in 3D object databases," *ACM Computing Surveys*, vol. 37, no. 4, pp. 345–387, 2005.

[11] Q. Zhang, J. Jia, and H. Li, "A GPU based 3D object retrieval approach using spatial shape information," in *Proceedings of the IEEE International Symposium on Multimedia (ISM '10)*, pp. 212–219, Taichung, Taiwan, December 2010.

[12] D. V. Vranic and D. Saupe, "3D model retrieval," in *Proceedings of the Spring Conference on Computer Graphics and its Applications (SCCG '00)*, 2000.

[13] J. W. H. Tangelder and R. C. Veltkamp, "A survey of content based 3D shape retrieval methods," *Multimedia Tools and Applications*, vol. 39, no. 3, pp. 441–471, 2008.

[14] T. Funkhouser, P. Min, M. Kazhdan et al., "A search engine for 3D models," *ACM Transactions on Graphics*, vol. 22, no. 1, pp. 83–105, 2003.

[15] K. Guo, W. Liu, and H. Song, "A novel mechanical component retrieval approach based on differential moment," *Advances in Mechanical Engineering*, vol. 2013, Article ID 401846, 5 pages, 2013.

[16] D. V. Vranić, D. Saupe, and J. Richter, "Tools for 3D-object retrieval: karhunen-Loeve transform and spherical harmonics," in *Proceedings of the IEEE fourth Workshop on Multimedia Signal Processing*, pp. 293–298, Cannes, France, October 2001.

[17] D. Y. Chen, X.-P. Tian, Y.-T. Shen, and M. Ouhyoung, "On visual similarity based 3D model retrieval," in *Computer Graphics Forum*, 2003.

[18] M. Kazhdan, "An approximate and efficient method for optimal rotation alignment of 3D models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 7, pp. 1221–1229, 2007.

[19] M. Chaouch and A. Verroust-Blondet, "A novel method for alignment of 3D models," in *Proceedings of the IEEE International Conference on Shape Modeling and Applications (SMI '08)*, pp. 187–195, Stony Brook, NY, USA, June 2008.

[20] M. Martinek and R. Grosso, "Optimal rotation alignment of 3D objects using a GPU-based similarity function," *Computers and Graphics (Pergamon)*, vol. 33, no. 3, pp. 291–298, 2009.

[21] D. V. Vranić, *3D model retrieval [Ph.D. thesis]*, University of Leipzig, Leipzig, Germany, 2004.

[22] J. Podolak, P. Shilane, A. Golovinskiy, S. Rusinkiewicz, and T. Funkhouser, "A planar-reflective symmetry transform for 3D shapes," in *Proceedings of the ACM (SIGGRAPH '06)*, pp. 549–559, August 2006.

[23] P. Daras, D. Zarpalas, D. Tzovaras, and M. G. Strintzis, "Efficient 3-D model search and retrieval using generalized 3-D radon transforms," *IEEE Transactions on Multimedia*, vol. 8, no. 1, pp. 101–114, 2006.

[24] J.-L. Shih and H.-Y. Chen, "A 3D model retrieval approach using the interior and exterior 3D shape information," *Multimedia Tools and Applications*, vol. 43, no. 1, pp. 45–62, 2009.

[25] P. Daras and A. Axenopoulos, "A 3D shape retrieval framework supporting multimodal queries," *International Journal of Computer Vision*, vol. 89, no. 2-3, pp. 229–247, 2010.

[26] P. Pospíchal, J. Jaros, and J. Schwarz, "Parallel genetic algorithm on the cuda architecture," in *Applications of Evolutionary Computation*, pp. 442–451, Springer, New York, NY, USA, 2010.

[27] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The princeton shape benchmark," in *Proceedings of the Shape Modeling International (SMI '04)*, pp. 167–178, June 2004.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 480954, 6 pages
http://dx.doi.org/10.1155/2013/480954

Research Article

Optimal Parameter Estimation for Muskingum Model Using a CSS-PSO Method

S. Talatahari,[1] R. Sheikholeslami,[2] B. Farahmand Azar,[1] and H. Daneshpajouh[3]

[1] *Department of Civil Engineering, University of Tabriz, Tabriz, Iran*
[2] *Department of Civil and Environmental Engineering, Amirkabir University of Technology, Tehran, Iran*
[3] *Marand Faculty of Engineering, University of Tabriz, Tabriz, Iran*

Correspondence should be addressed to S. Talatahari; siamak.talat@gmail.com

Received 22 August 2013; Accepted 23 September 2013

Academic Editor: Shengyong Chen

Limited availability of hydrologic data is a major hurdle for implementation of detailed hydrologic models. In cases where available data is limited, simple hydrologic models such as linear Muskingum model consisting of a minimum number (one or two) of model parameters are more desirable. As an alternative to the conventional mathematical approaches, this paper applies a new hybrid metaheuristic algorithm based on charged system search and particle swarm optimization for identifying the parameters of the linear Muskingum model. In order to evaluate the new algorithm, a numerical example is utilized and the results are compared to those of other algorithms. The results reveal the performance of the algorithm to optimize parameter estimation of the Muskingum model.

1. Introduction

Charged system search (CSS) is a relatively new metaheuristic optimization algorithm inspired by the governing laws of electrical physics and the Newtonian mechanics. In electrical physics, the electric charge can generate the electric field and exerts a force on other electrically charged objects. The electric field surrounding a point charge is specified by the laws of Coulomb and Gauss. Utilizing these principles, the CSS algorithm defines a number of solution candidates each of which is called charged particle (CP) and is treated as a charged sphere. Each CP can exert an electrical force on the other agents (CPs). These forces can change the position of other CPs according to Newton's second law. Finally, considering the Newtonian mechanics, the new positions of CPs are determined [1, 2]. Also a hybrid CSS and PSO (particle swarm optimization) algorithm is proposed by Kaveh and Talatahari [3]. The hybrid CSS-PSO has shown good performance in solving optimization problems in different areas such as optimal design of large-scale structures [4] and optimal design of laminated composite structures [5].

For the first time we apply this hybrid algorithm for optimal parameter estimation of Muskingum model. The Muskingum model is a popular model for flood routing which is basically an extension of the reservoir routing methods taking into account that the storage (S) in a channel is a function of both the inflow (I) and the outflow (O). This means that in addition to the prism storage there is a (positive or negative) wedge storage. However, the application of the model still suffers from an absence of an efficient method for parameter estimation. There are various mathematical techniques which have been used for estimating the parameters of the Muskingum model. These mathematical methods use gradient information to search the solution space near an initial starting point [6]. In order to evaluate the proposed method, a numerical example is utilized and the results are compared to those of other algorithms.

2. Charged System Search Algorithm

2.1. General Aspects. The charged system search (CSS) algorithm is a relatively novel metaheuristic based on the Coulomb and Gauss laws from electrical physics and the governing laws of motion from the Newtonian mechanics [1]. This algorithm can be considered as a multiagent approach,

where each agent is a charged particle (CP). Each CP is considered as a charged sphere with radius a, having a uniform volume charge density and is equal to

$$q_i = \frac{\text{fit}(i) - \text{fitworst}}{\text{fitbest} - \text{fitworst}} \quad i = 1, 2, \ldots, N, \tag{1}$$

where fitbest and fitworst are the best and the worst fitness of all the particles, fit(i) represents the fitness of the agent i, and N is the total number of CPs.

CPs can impose electric forces on the others, and their magnitude for the CP located in the inside of the sphere is proportional to the separation distance between the CPs, and for a CP located outside the sphere is inversely proportional to the square of the separation distance between the particles. The kind of the forces can be attractive or repelling and it is determined by using ar_{ij}, the kind of force parameter, as

$$ar_{ij} = \begin{cases} +1 & \text{w.p. } k_t \\ -1 & \text{w.p. } 1 - k_t, \end{cases} \tag{2}$$

where ar_{ij} determines the type of the force, +1 represents the attractive force and –1 denotes the repelling force, and k_t is a parameter to control the effect of the kind of force. Therefore, the resultant force is redefined as

$$\begin{gathered} \mathbf{F}_j = q_j \sum_{i, i \neq j} \left(\frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2 \right) ar_{ij}\, p_{ij} \left(\mathbf{X}_i - \mathbf{X}_j \right) \\ j = 1, 2, \ldots, N, \; i_1 = 1, i_2 = 0 \Longleftrightarrow r_{ij} < a \\ i_1 = 0, i_2 = 1 \Longleftrightarrow r_{ij} \geq a, \end{gathered} \tag{3}$$

where $\mathbf{F}_j$ is the resultant force acting on the jth CP; r_{ij} is the separation distance between two charged particles defined as

$$r_{ij} = \frac{\left\| \mathbf{X}_i - \mathbf{X}_j \right\|}{\left\| \left(\mathbf{X}_i + \mathbf{X}_j \right) / 2 - \mathbf{X}_{\text{best}} \right\| + \varepsilon}, \tag{4}$$

where $\mathbf{X}_i$ and $\mathbf{X}_j$ are the positions of the ith and jth CPs, respectively, $\mathbf{X}_{\text{best}}$ is the position of the best current CP, and ε is a small positive number to avoid singularity. p_{ij} determines the probability of moving each CP toward the others as

$$p_{ij} = \begin{cases} 1 & \dfrac{\text{fit}(i) - \text{fitbest}}{\text{fit}(j) - \text{fit}(i)} > \text{rand} \vee \text{fit}(j) < \text{fit}(i) \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

At the movement stage, each CP moves toward its new position under the action of the resultant forces and its previous velocity as

$$\begin{gathered} \mathbf{X}_{j,\text{new}} = \text{rand}_{j1} \cdot k_a \cdot \frac{\mathbf{F}_j}{m_j} \cdot \Delta t^2 \\ + \text{rand}_{j2} \cdot k_v \cdot \mathbf{V}_{j,\text{old}} \cdot \Delta t + \mathbf{X}_{j,\text{old}} \\ \mathbf{V}_{j,\text{new}} = \frac{\mathbf{X}_{j,\text{new}} - \mathbf{X}_{j,\text{old}}}{\Delta t}, \end{gathered} \tag{6}$$

where k_a is the acceleration coefficient; k_v is the velocity coefficient to control the influence of the previous velocity and here both are equal to 0.5; and rand_{j1} and rand_{j2} are two random numbers uniformly distributed in the range (0, 1). If each CP moves out of the search space, its position is corrected using the harmony search-based handling approach as described in [7]. In addition, to save the best design, a memory (charged memory) is utilized.

2.2. Hybrid PSO and CSS. The particle swarm optimization (PSO) involves a number of particles, which are initialized randomly in the space of the design variables. These particles fly through the search space and their positions are updated based on the best positions of individual particles and the best position among all particles in the search space which corresponds to a particle with the smallest weight. Also considering the position of a particle selected randomly from the swarm (passive congregation) can improve the performance of the algorithm.

The update moves a particle by adding a change velocity V_j^{k+1} to the current position X_k^j as follows:

$$\begin{gathered} X_j^{k+1} = X_j^k + V_j^k, \\ V_j^{k+1} = \omega V_j^k + c_1 r_1 \left(P_j^k - X_j^k \right) \\ + c_2 r_2 \left(P_g^k - X_j^k \right) + c_3 r_3 \left(R_j^k - X_j^k \right), \end{gathered} \tag{7}$$

where ω is an inertia weight to control the influence of the previous velocity; r_1, r_2, and r_3 are three random numbers uniformly distributed in the range of (0, 1); c_1 and c_2 are two acceleration constants; c_3 is the passive congregation coefficient; P_j^k is the best position of the jth particle up to iteration k; P_g^k is the best position among all particles in the swarm up to iteration k; and R_k is a particle selected randomly from the swarm. In order to increase the exploration ability, in this paper the velocity of the particles is defined as

$$\begin{gathered} V_j^{k+1} = \omega V_j^k + c_1 r_1 \left(P_j^k - X_j^k \right) + c_2 r_2 \left(P_g^k - X_j^k \right) \\ + c_3 r_3 \left(R_j^k - X_j^k \right) + c_4 r_4 \left(Rd_j^k - X_j^k \right), \end{gathered} \tag{8}$$

where c_4 is the exploration coefficient; r_4 is a uniformly distributed random number in the range of (0,1); and Rd^k is a vector generated randomly from the search domain.

Compared to PSO, the CSS algorithm can be considered as a generalized form of the PSO containing its superiorities, but avoiding its disadvantages. Both are population-based algorithms and find optimum solutions by changing the position of the agents. However, the movement strategies are different for the CSS and PSO. The PSO algorithm utilizes the local best and the global best to determine the direction of the movement, while the CSS approach uses the governing laws from electrical physics and the Newtonian mechanics to determine the amount and the direction of a charged particle's movement. CSS can distinguish the local search and the global search phases and utilizes suitable relationships in these phases resulting in a good balance

between the exploration and exploitation, while the greatest disadvantage of the PSO approach is the existence of some difficulties in controlling the balance between the exploration and exploitation.

In the present hybrid algorithm [3], the advantage of the PSO consisting of utilizing the local best and the global best is added to the CSS algorithm. The **CM** updating process is defined as follows:

$$\mathbf{CM}_{i,\text{new}} = \begin{cases} \mathbf{CM}_{i,\text{old}} & \text{fit}\left(\mathbf{X}_{i,\text{new}}\right) \geq \text{fit}\left(\mathbf{CM}_{i,\text{old}}\right) \\ \mathbf{X}_{i,\text{new}} & \text{fit}\left(\mathbf{X}_{i,\text{new}}\right) \geq \text{fit}\left(\mathbf{CM}_{i,\text{old}}\right). \end{cases} \tag{9}$$

Considering the above new **CM**, the electric forces generated by agents are modified as

$$\begin{aligned} \mathbf{F}_j = &\sum_{i\in S_1}\left(\frac{q_i}{a^3}r_{ij}\cdot i_1 + \frac{q_i}{r_{ij}^2}\cdot i_2\right) ar_{ij}\, p_{ij}\left(\mathbf{CM}_{i,\text{old}} - \mathbf{X}_j\right) \\ &+ \sum_{i\in S_2}\left(\frac{q_i}{a^3}r_{ij}\cdot i_1 + \frac{q_i}{r_{ij}^2}\cdot i_2\right) ar_{ij}\, p_{ij}\left(\mathbf{X}_i - \mathbf{X}_j\right), \end{aligned} \tag{10}$$

where subtitles S_1 and S_2 denote two sets of the numbers which determine the number of the agents utilized to calculate the resultant force by employing the agents sorted in the **CM** and the current agents positions, respectively. If the coefficient k_i is defined as

$$k_i = \left(\frac{q_i}{a^3}r_{ij}\cdot i_1 + \frac{q_i}{r_{ij}^2}\cdot i_2\right) ar_{ij}\, p_{ij}, \tag{11}$$

then the resultant force formulae can be simplified as

$$\begin{aligned} \mathbf{F}_j = &k_1\left(\mathbf{CM}_{g,\text{old}} - \mathbf{X}_j\right) + k_2\left(\mathbf{CM}_{j,\text{old}} - \mathbf{X}_j\right) \\ &+ \sum_{i\in S_1} k_i\left(\mathbf{CM}_{i,\text{old}} - \mathbf{X}_j\right) + \sum_{i\in S_2} k_i\left(\mathbf{X}_i - \mathbf{X}_j\right), \end{aligned} \tag{12}$$

where the subtitle g denotes the number of the stored so far good position among all CPs. Therefore the first term directs the agents towards the global best position. When $i = j$, then the $\mathbf{CM}_{i,\text{old}}$ is treated similar to $\mathbf{P}_i^k$ in the PSO as considered in the second term of the above equation. This will direct the agents towards the local best [4, 5].

3. Determining the Muskingum Model Parameters

3.1. Linear Muskingum Model. Hydrologic modeling plays a key role in water resources planning, development, and management. Limited availability of hydrologic data is a major hurdle for implementation of detailed hydrologic models. In cases where available data is limited, simple hydrologic models consisting of a minimum number (one or two) of model parameters are more desirable [8]. A simple conceptual model like the Muskingum model is very effective in the estimation of the hydrograph of the rise and fall of a river at any given point on the river during the course of a flood event.

The problem is solved by the techniques of flood routing, which is a process for calculating the shape of a flood wave along a river channel. McCarthy [9] analyzed the data from the Muskingum River in Ohio and found certain relationships: storage is proportional to outflow; and storage is also proportional to the difference between inflow and outflow [10]. In the linear Muskingum model, the following continuity and storage equations are used:

$$\begin{gathered} \frac{dS_t}{dt} \approx \frac{\Delta S_t}{\Delta t} = I_t - O_t, \\ S_t = K\left[xI_t - (1-x)\,O_t\right], \end{gathered} \tag{13}$$

where S_t represents the storage volume within the reach at time t (L^3); I_t and O_t are rates of the inflow and outflow at time t ($\mathrm{L}^3\,\mathrm{T}^{-1}$), respectively; K is the storage-time constant for the river reach (T), which has a value close to the flow travel time through the river reach; x is the dimensionless weighting factor commonly varying between 0.00 and 0.30 for stream channels and between 0.00 and 0.50 for reservoir storage; and Δt represents time step (T).

Choosing a suitable time interval for the routing period, the continuity equation in its finite difference form becomes

$$S_{i+1} - S_i = \frac{\left(I_i + I_{i+1}\right)\Delta t}{2} - \frac{\left(O_i + O_{i+1}\right)\Delta t}{2} \tag{14}$$

using the linear Muskingum model:

$$O_{i+1} = C_1 I_i + C_2 I_{i+1} + C_3 O_i, \quad (i = 1, 2, 3, \ldots), \tag{15}$$

where

$$\begin{gathered} C_1 = \frac{2Kx + \Delta t}{\Delta t + 2K - 2Kx}, \qquad C_2 = \frac{-2Kx + \Delta t}{\Delta t + 2K - 2Kx}, \\ C_3 = \frac{2K - 2Kx - \Delta t}{\Delta t + 2K - 2Kx} \end{gathered} \tag{16}$$

in which O_i and I_i represent the outflow and inflow discharges at time t_i, respectively; and C_1, C_2, and C_3 are the Muskingum parameters where $C_1 + C_2 + C_3 = 1$. The routing procedure consists of the following steps.

Step 1. Assume the values of two parameters (K and x).

Step 2. Calculate the coefficients C_1, C_2, and C_3 using (16).

Step 3. Compute the magnitude of the outflow at the next time (O_{i+1}) using (15).

Step 4. Repeat Step 3 for all times.

In the conventional method, the parameters K and x in the linear Muskingum model are graphically estimated by a trial and error procedure. After assuming x, the values of $[XI_t + (1 - X)O_t]$ are computed using observed data in both upstream and downstream and then they are plotted against S_t. The value of x that minimizes the width of the plotted loop can be chosen as the correct value of x, and the slope of the straight line fitted through the loop gives K [11]. There are some disadvantages of the conventional trial and error method.

(a) The satisfactory results obtained often need calculations of several times, rather than one time.

(b) Because two parameters K, x are got through trying and error artificiality, the introduction of human errors will be unavoidable.

(c) The results cannot be proved theoretically optimum solution of the Muskingum parameters and therefore cannot ensure the global optimum of C_1, C_2, and C_3.

(d) This method not only increases the manual workload but also declines the accuracy of solution and it is difficult to programming.

Another traditional approach for determining Muskingum model parameters is the least square method (LSM). The LSM is a scientific solution for the Muskingum model under the condition of sufficient historical data. Because of its reliability and practicality, this method has been widely accepted. The biggest shortcoming of the LSM is the heavy dependence on historical data. If the historical data are complete and the time intervals are nearly equally distributed, the results of this approach will be good. On the contrary, if the historical data are incomplete, time intervals are too long or unequal, the results will be error greatly.

3.2. An Optimization Approach. In (15), C_1, C_2, and C_3 are hydrologic parameters of the model. The main challenge is to find a set of parameters that give a representative response. These parameters are determined using optimization techniques by obtaining a good match between the observed and calculated discharge. Choosing of a suitable objective function plays an important role in optimization problems. Here, we use the following objective function which is based on minimizing the sum of squares of deviations between observed and computed values of outflow:

$$\mathrm{SSQ} = \sum_{j=1}^{N_o} \left(O_{oj} - O_{cj}\right)^2, \tag{17}$$

where O_{oj} is observed value of the outflow at time t_j; O_{cj} is the computed value of the outflow at time t_j; j is an index varying from one to N_o, where N_o is the number of hydrograph ordinates of the observed hydrograph. Thus, the optimal parameter estimation for Muskingum model may be expressed as follows:

Find

$$\{C_1, C_2, C_3\} \tag{18}$$

To minimize

$$\mathrm{SSQ}\left(C_1, C_2, C_3\right) \tag{19}$$

Subject to

$$C_1 + C_2 + C_3 = 1. \tag{20}$$

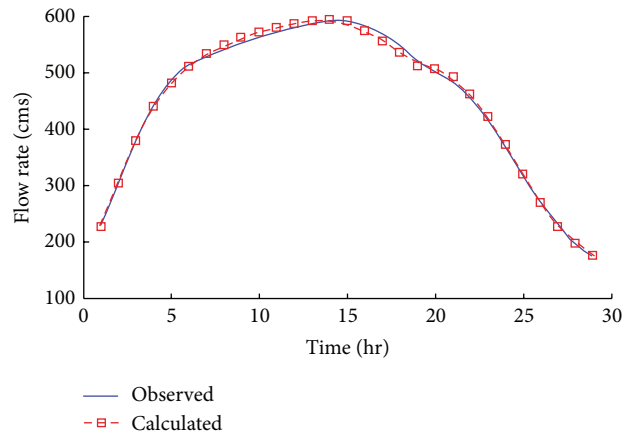


Figure 1: Observed and routed hydrographs.

4. Example of Hydrologic Parameter Estimation for Muskingum Model

To investigate the performance of applying the new algorithm to solve the parameter estimation problem of the linear Muskingum model, a typical problem is used as an example. The algorithms were implemented using MATLAB to run on a computer with an Intel Core i5 CPU, 2.53 GHz processor, and 3.00 GB RAM; also the penalty method is used to handle the constraint. In this paper, the model is applied to the south canal between Chenggou and Linqing rivers in China. The length of the river course (from Chenggou to Linqing) of South Canal in Haihe Basin reaches 83.8 km. There is no significant effect of the branches. The time interval in the calculation is taken as 12 hr. The data set from [13] is considered for illustration purpose. This example has been studied previously by Wang et al. [14, 15], Zahn and Xu [16], and Yang et al. [17] for testing different parameter estimation methodologies. Also Zhengxiang and Ling [12] applied and compared several intelligent algorithms on Muskingum routing model using this example. Therefore, the performance of the proposed algorithm can effectively be compared with the previously reported results obtained with this example.

In Tables 1 and 2, the optimum parameters and the objective function of the solutions obtained by the least squares method (LSM), GA, SA, PSO [12], ICA [6], and the proposed CSS-PSO are presented. For the proposed algorithm, a population of 10 particles is used. The results show that the SSQ is attained using new algorithm. It has been demonstrated that the new algorithm gets better results than those of other methods. The computed outflow hydrograph and the observed outflow hydrograph are shown in Figure 1. The plot depicts that the resulting outflow hydrograph, using the parameters estimated from the proposed method, closely follows the observed outflows.

5. Conclusion

In this research a powerful algorithm is developed by combining the CSS and PSO. The new algorithm in the CSS

Table 1: Comparison of best SSQs from various methods.

	SSQ	C_1	C_2	C_3
LSM [12]	1311.28	0.0843	0.4626	0.4531
GA [12]	1046.69	0.05476	0.45241	0.49283
SA [12]	1046.69	0.0547522	0.4524160	0.4928318
PSO [12]	1046.83	0.0547561	0.4524157	0.4928282
ICA [6]	1031.28	0.0100000	0.4774254	0.5116732
This study (CSS-PSO)	1020.22	0.0428869	0.4599482	0.4963921

Table 2: Comparison of the observed and computed outflows (cm).

I (inflow)	Q_o (observed outflow)	Q_c (computed outflow)
261	228	228.00
389	300	303.29
462	382	379.73
505	444	440.58
525	490	481.83
543	513	511.44
556	528	532.89
567	543	549.16
577	553	562.31
583	564	572.02
587	573	578.94
595	581	586.22
597	588	591.10
597	594	593.61
589	592	591.17
556	584	574.44
538	566	556.44
516	550	536.62
486	520	512.04
505	504	507.29
477	483	492.86
429	461	462.43
379	420	422.26
320	368	373.05
263	318	319.87
220	271	271.24
182	234	227.79
167	193	197.69
152	178	175.21

formulation introduces the PSO concept of local best and global best. The main advantages of the PSO consisting of directing the agents toward the global best (obtained by the swarm) and the local best (obtained by the agent itself) are added to the CSS algorithm to improve its performance. In the present approach, similar to the original CSS, each agent is affected by other agents considering the governing laws of electrical physics. However, the kind of the forces can be repulsive and attractive. For the proposed algorithm, when the calculations of the amount of forces are completed for the CPs and the new locations of agents are determined, the **CM** updating is performed.

Here, the hybrid CSS-PSO method has been proposed in order to improve the computational accuracy for parameter estimation of Muskingum model. The Muskingum method ignores the momentum equation and is based solely on the continuity equation. The proposed optimization approach found the best value of parameters in terms of the minimal sum of the square deviation between the observed and routed outflows. The performance of this approach was compared with those of other common methods such as GA, SA, PSO, and ICA. The results demonstrate that the new algorithm can achieve a high degree of accuracy to estimate the parameters. This method appears to offer good applicability in the hydrology field and further applications should be explored.

References

[1] A. Kaveh and S. Talatahari, "A novel heuristic optimization method: charged system search," *Acta Mechanica*, vol. 213, no. 3-4, pp. 267–289, 2010.

[2] A. Kaveh and S. Talatahari, "Optimal design of skeletal structures via the charged system search algorithm," *Structural and Multidisciplinary Optimization*, vol. 41, no. 6, pp. 893–911, 2010.

[3] A. Kaveh and S. Talatahari, "Hybrid charged system search and particle swarm optimization for engineering design problems," *Engineering Computations*, vol. 28, no. 4, pp. 423–440, 2011.

[4] A. Kaveh and S. Talatahari, "Optimization of large-scale truss structures using modified charged system search," *International Journal of Optimization in Civil Engineering*, vol. 1, no. 1, pp. 15–28, 2011.

[5] A. Kaveh, B. H. Soudmand, and R. Sheikholeslami, "Optimal design of laminated composite structures via hybrid charged system search and particle swarm optimization," *Asian Journal of Civil Engineering*, vol. 14, no. 4, pp. 587–604, 2013.

[6] A. Tahershamsi and R. Sheikholeslami, "Optimization to identify Muskingum model parameters using imperialist competitive algorithm," *International Journal of Optimization in Civil Engineering*, vol. 1, no. 3, pp. 473–482, 2011.

[7] A. Kaveh and S. Talatahari, "A charged system search with a fly to boundary method for discrete optimum design of truss structures," *Asian Journal of Civil Engineering*, vol. 11, no. 3, pp. 277–293, 2010.

[8] M. M. Ahmad, A. R. Ghumman, S. Ahmad, and H. N. Hashmi, "Estimation of a unique pair of nash model parameters: an optimization approach," *Water Resources Management*, vol. 24, no. 12, pp. 2971–2989, 2010.

[9] G. T. McCarthy, "The unit hydrograph and flood routing," in *Proceedings of the North Atlantic Division Conference*, U.S. Army Corps of Engineers, New London, Conn, USA, 1938.

[10] Z. W. Geem, "Parameter estimation of the nonlinear Muskingum model using parameter-setting-free harmony search," *Journal of Hydrologic Engineering*, vol. 16, no. 8, pp. 684–688, 2011.

[11] R. Barati, "Parameter estimation of nonlinear Muskingum models using nelder-mead simplex algorithm," *Journal of Hydrologic Engineering*, vol. 16, no. 11, pp. 946–954, 2011.

[12] Y. Zhengxiang and K. Ling, "Application and comparison of several intelligent algorithms on Muskingum routing model," in *Proceedings of the 2nd IEEE International Conference on Information and Financial Engineering (ICIFE '10)*, pp. 910–914, Chongqing, China, September 2010.

[13] G. J. Qu, "Parameter estimation for Muskingum models," *Journal of Hydrology*, vol. 36, no. 3, pp. 40–43, 1977.

[14] W. Wang, Z. Xu, L. Qiu, and D. Xu, "Hybrid chaotic genetic algorithms for optimal parameter estimation of Muskingum flood routing model," in *Proceedings of the IEEE International Joint Conference on Computational Sciences and Optimization (CSO '09)*, pp. 215–218, Hainan, China, April 2009.

[15] W. Wang, Y. Kang, and L. Qiu, "Optimal parameter estimation for Muskingum model using a modified particle swarm algorithm," in *Proceedings of the 3rd IEEE International Joint Conference on Computational Sciences and Optimization (CSO '10)*, pp. 153–156, Anhui, China, May 2010.

[16] S. Zhan and J. Xu, "Application of ant colony algorithm to parameter estimation of Muskingum routing model," *Journal of Natural Disasters*, vol. 14, no. 5, pp. 20–24, 2005.

[17] X. Yang, J. Jin, Z. Chen, and Y. Wei, "A new approach for parameter estimation of Muskingum routing model," *Journal of Catastrophology*, vol. 13, pp. 1–6, 1998.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 610604, 9 pages
http://dx.doi.org/10.1155/2013/610604

Research Article

Recommendation in Motion: Intelligent Hypertouch Garment Design

Shuang Liang,[1] George Baciu,[2] Rong-Hua Li,[3] Jinyuan Jia,[1] and Jianhua Zhang[4]

[1] *School of Software Engineering, Tongji University, Shanghai 201804, China*
[2] *Department of Computing, The Hong Kong Polytechnic University, Hong Kong*
[3] *Department of System Engineering & Engineering Management, The Chinese University of Hong Kong, Hong Kong*
[4] *College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310014, China*

Correspondence should be addressed to Jinyuan Jia; jyjia@tongji.edu.cn

Received 9 August 2013; Accepted 23 September 2013

Academic Editor: Shengyong Chen

Intelligent CAD garment design becomes more and more popular by attracting the attentions from both manufacturers and professional stylists. The existing garment CAD systems and clothing simulation software fail to provide user-friendly interfaces as well as dynamic recommendation during the garment creation process. In this paper, we propose an intelligent hypertouch garment design system, which dynamically predicts the possible solutions along with the intelligent design procedure. User behavioral information and dynamic shape matching are used to learn and predict the desired garment patterns. We also propose a new hypertouch concept of gesture-based interaction for our system. We evaluate our system with a prototype platform. The results show that our system is effective, robust, and easy to use for quick garment design.

1. Introduction

During the last decade, garment design, modeling, and animation have been identified as being among the most popular topics in the fields of CAD and computer graphics. Traditional garment design usually requires pattern technologists (tailors) to create 2D patterns from the designer's sketches, which are then cut out and sewn together to form a complete piece of dress. CAD systems and software [1] are highly involved in clothing manufacturing to model these 2D clothes patterns according to the specifications from the designers and tailors.

However, manipulating a CAD system requires extensive learning effort. Most commercial garment design systems are based on rigid shape creation and editing, such as B-Spline and Bézier Curve. It is neither flexible to perform the operation nor efficient to carry out validity assessment. Moreover, it is often required to obtain a complete shape input before the CAD system starts to process, resulting in efficiency being lost in the garment CAD progress. It would be better to have a proactive clothes shape recommendation mechanism that captures the knowledge of user preferences and predicts and suggests suitable solutions in such a dynamic drawing environment. This is the first motivation of our work.

Our second motivation is the fast development of multitouch technology [2]. Current CAD platforms for garment design are mainly conceived for 2D modeling for geometric clothes shapes but generally do not provide high-level interactive environment with aesthetic and functional features. With the emergence of electronic touch-sensing equipments, it is now possible to provide designers with a more intuitive and flexible means of interaction. The multitouch systems allow people to interact with the system using multiple fingers simultaneously. More information is processed, and higher degrees of freedom are experienced than in mouse-based operations. This in turn reduces the time and effort overhead for prior training. How to build an efficient and easy-to-use garment design platform becomes one of the main challenges in the domain of textile garment production. We propose a concept of hypertouch interaction to enrich the diversity and functionality of current touch-based interfaces.

Contributions. We propose a proactive recommendation system that returns possible clothes shapes along with

the designing progress based on user behavior patterns. The work we present in this paper has three main contributions, which are given as follows.

(1) *User Behavior Pattern Extraction.* We extract the behavior patterns from the apparel design database to capture user preferences. A domain knowledge base with these user behavior patterns is constructed accordingly. We employ the user behavior tree from our previous work [3] to model these user patterns. Different drawing habits with respect to both spatial and temporal properties are taken into consideration.

(2) *Dynamic Clothes Shape Recommendation.* We introduce a new feature of dynamic recommendation, which is also called recommendation in motion, to the current apparel design systems. We provide a proactive mechanism, which dynamically returns fitting shapes as the designer inputting clothes contours. This speeds up the apparel design process to a large extent. The selection of the matching shapes is determined by a partial matching algorithm between the incomplete input shape and the patterns in the knowledge base.

(3) *Nonrigid Hypertouch Interaction.* We propose a new hypertouch interaction concept for apparel design. Compared to the existing touch-based systems, the hypertouch interface gives the user higher degrees of freedom as well as more information throughput. A flexible and natural interface is developed to benefit the apparel design task towards a more efficient and user-friendly direction.

2. Related Work

Recent touch-sensing techniques [2] enable people to interact with complex systems through multipoint control and have become the focus of a great deal of research and commercial activities [4]. These techniques support gesture-based interaction [5, 6], with or without the requirement of direct contact with the touch screen or the so-called hypertouch [7], to provide effective and practical manipulations or controls of the objects.

Wu and Balakrishnan [6] explored a variety of gestural interaction techniques with multifingers or whole hands. Their techniques are towards tabletop displays as if interacting with physical objects on real tables. ShadowGuides [8] is an interactive system for multitouch and hand gestures with in situ learning. Wobbrock et al. [9] presented a method to design tabletop gestures that relies on eliciting gestures from nontechnical users by first portraying the effect of a gesture and then asking users to perform its cause.

There are some gestural systems developed to facilitate the processing of clothing manipulation. Igarashi and Hughes [10] proposed a technique for mapping sketching clothes marks onto a 3D model by building up correspondence between 2D sketches and 3D models. In [11], authors proposed a garment creation and modeling system with sketches. The operations are performed by stroke symbols designed especially for different editing operations, such as break, corner detection, and folding.

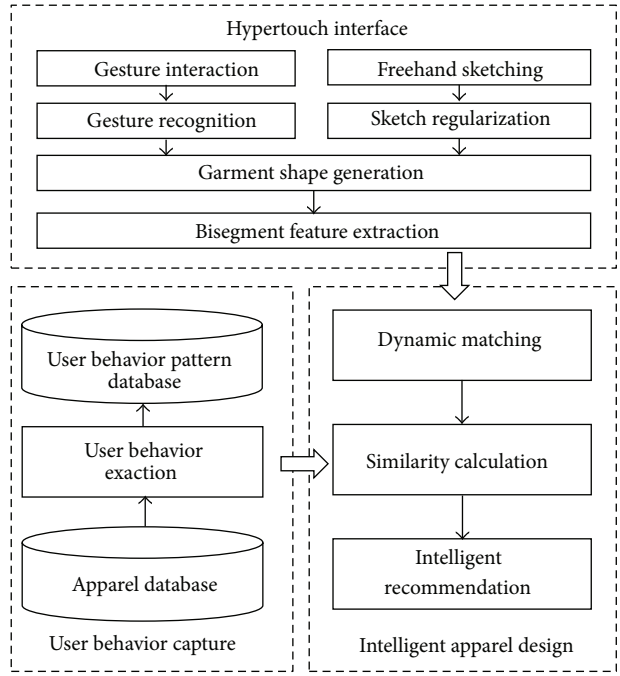


Figure 1: The architecture of our intelligent hypertouch recommendation system for apparel design.

These systems carry out clothing design tasks on virtual human models, and do not provide domain knowledge during the design process. Moreover, all operations correspond to a single-finger dragging and moving action of different paths and patterns, while a multitouch concept is not involved.

3. System Architecture

In this paper, we present an intelligent recommendation system for garment design. The propose method provides an easy-to-use and flexible touching interface, as well as dynamic shape prediction based on user behavior patterns. Figure 1 demonstrates the entire system architecture and framework of our proposed method.

Three phases are involved in our proposed system, including (1) user behavior capture, (2) hypertouch interaction, and (3) intelligent apparel design.

For the first phase of our system, a variety of user behavior patterns are captured by constructing a drawing pattern database offline. We first invite designers to perform apparel design to generate clothes shapes. The garment creation process is recorded and the drawing behavior patterns from designers and tailors are extracted and retrieved. These user drawing behavior patterns are formulated into a tree-structured model, which is called the user behavior tree (UBT) in our previous work [3]. A knowledge base storing the UBTs of the designers and tailors is constructed accordingly.

For the second phase, a hypertouch interface is provided to perform the apparel design task. By using the interface, the garment shapes are depicted freely and quickly. Both freehand sketching and gesture interaction are supported. New shape inputs and commands are created by designers with stylus or fingers on our tabletop display device. The raw input data are first regularized and recognized. Geometric features and topological features of the shape are then extracted. In this paper, bisegment descriptors [12] are used to model these shape features.

The third phase is online dynamic recommendation. This is a google-like processing. Possible matched shapes are proactively returned to the user to provide suggestions and reference according to domain knowledge and user preferences. A partial matching algorithm proposed in our previous work [13] is used to dynamically return the possible shapes with maximal similarities to the current input drawing in real time. We develop a prototype system to evaluate the effectiveness, efficiency, and user friendliness of the proposed method.

4. Hypertouch Interaction

In this paper, we propose a new type of interaction called hypertouch (HT) interaction. The hypertouch interaction is an extension of the existing touch-based interfaces [2]. It exploits richer ways of interactions and puts them into a better and broader manner of usage.

In this section, we first define the concept of hypertouch interaction. Second, we define the elementary items forming hypertouch interaction and discuss their characteristics.

4.1. The Hypertouch Concept. The richness and diversity of user interfaces is highly related to the richness of degrees of freedom (DOF), particularly continuous DOF. Typical touch-based interfaces are largely based on the moving fingers on the 2D touch-sensitive surfaces. Direct touches on the surfaces, either with a stylus or fingers, are required in order to perform practical controls on the objects. Direct touching is required, either with a stylus or fingers, on the surface in order to perform practical controls on the objects. This results in 2 degrees of freedom (DOF) for each finger touch and places a 2D spatial constraint on the interaction. Sometimes it is not adequate to capture the type of richness of input that we encounter in our interaction world.

In order to enable more channels for communication, we propose the concept of hypertouch interaction, which is given in Definition 1.

Definition 1 (hypertouch interaction). Hypertouch interaction is a manner of human-computer interaction with a touch screen, in which each touch event gives two or more degrees of freedom (DOF).

By providing higher DOF, the hypertouch interaction opens up the diversity of touch functionalities. It alleviates the 2D spatial constraint for current touch interfaces that the user has to actually touch the screen to operate. In other words, it extends the touch actions into space-immersed manipulations with higher dimensions. The conventional 2D touch-based interfaces are enriched with larger freedom and flexibility in a broader spatial area.

The common 2D touch-based interfaces are the most popular and typical hypertouch systems. However, these touch-based systems are basically offering 2 DOF for each finger touch. In addition to the 2D touch interface, hypertouch interaction also happens in 2.5D and 3D spaces. Higher DOF allows the user to conduct the interaction more freely and flexibly. This generates friendly user experience and improves the working efficiency.

The 2.5D hypertouch interaction is also called zero touch or touch-free, which means the interaction takes place in an area very close to the touch surface (usually within 20 cm in distance). This activates the depth sensing feature of the operations, which provides a far higher potential for rich interaction than the current binary touch interaction.

The 3D hypertouch is also known as 3D gesture control or 3D depth sensing technique. It allows the user to interact with any display screen from any distance. Depth tracking devices, for example, 3D cameras and 3D gesture recognition, are required to perform the 3D hypertouch interaction.

The above spatial attributes are one of the most intrinsic characteristics of the hypertouch interaction that makes it more comprehensive than the current 2D touch-based interaction. It will be formalized and discussed in detail in the following subsection.

4.2. Characteristics of Hypertouch Interaction. The hypertouch interaction is composed of a series of hypertouch events under certain context. The diversity of a hypertouch event is determined by two factors as the multitouch threads and the gesture depth channel. Definition 2 gives the concept of hypertouch event.

Definition 2 (hypertouch event). A hypertouch event ω is a 2-tuple which defines the hypertouch action involving ν finger touches at distance z and is denoted as

$$\omega = \langle \nu, z \rangle, \quad \nu \in \mathfrak{R}^{+}, \ z \geq 0, \tag{1}$$

where ν is the number of touch threads and z is the gesture depth from the hands to the screen along the Z-axis.

Figure 2 shows the principle of hypertouch event, in which the two axes denote for the touch threads and the depth channel, respectively. As we can see from Figure 2, the interaction space is described by the above two dimensions. The flexibility (the red line) of gestures is enriched with the growing of distance to the screen and the number of the touches. This is because more touch points and broader interaction space provide more free and natural communication as is in our daily lives. On the other hand, high flexibility results in low efficiency and accuracy. Since additional time is needed for 2.5D and 3D interaction to track the hand position and perform 3D gesture recognition, the efficiency (the green line) of the interaction is decreased along with the increasing of the gesture depth. The goal of hypertouch event is to achieve a tradeoff between flexibility and efficiency. This

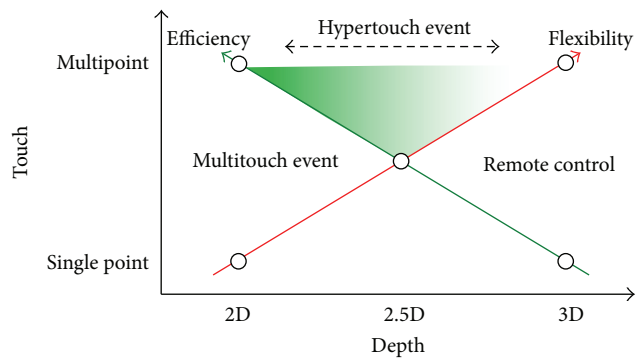


Figure 2: The principle of hypertouch event. The hypertouch event takes action in the upper subspace, while multitouch event and remote control are performed in the left and right subspace, respectively.

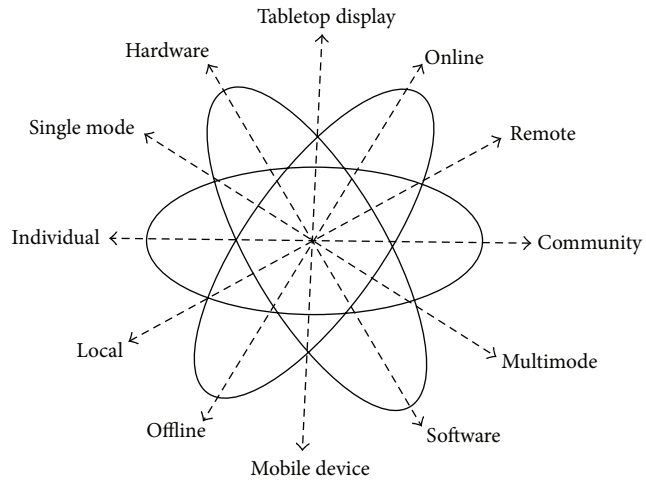


Figure 3: The illustration of various attributes in the hypertouch context from different aspects.

means the hypertouch event usually takes place in the upper subspace (the shadowed area) in order to generate satisfied user experiences. In other words, users tend to use multiple number of touches across multiple dimensions of spaces to achieve an optimal combination of flexibility and efficiency. This also explains why the current 2D multitouch interface is not enough to provide flexible and constraint-free gesture functionalities.

A hypertouch event and its corresponding context information consist of a scene of hypertouch. A set of hypertouch scenes consist of the whole hypertouch interaction space. The concept of hypertouch scene and interaction space are defined in Definitions 3 and 4, respectively.

Definition 3 (hypertouch scene). A scene δ is a scenario in which the users perform their hypertouch actions. A scene is composed of a $\langle$event, context$\rangle$ pair as

$$\delta = \langle \omega, \lambda \rangle, \quad \lambda \in \Lambda, \tag{2}$$

where ω is the hypertouch event and λ is the context of the scene from the context set Λ.

Definition 4 (hypertouch space). An interactive hypertouch space Ψ is consisted of a set of hypertouch scenes, which are aware of the user, performance, application, and environment. It is denoted as

$$\Psi^n = \{\delta_1, \delta_2, \ldots, \delta_n\}, \tag{3}$$

where δ stands for a specific scene.

The context information is taken into consideration in order to model the hypertouch scene more comprehensively. The context is related to the specific environmental parameters and attributes of the hypertouch event. It encodes the operating characteristics from multiple aspects, including time, location, number of users, interaction mode, software API, and hardware. Figure 3 shows an illustration of the multiple dimensions of the context information.

5. Intelligent Garment Recommendation

The intelligent garment design is a real-time interaction process between the designer and the computer. The ultimate goal of the system is to support dynamic recommendations along with the shape generation process. To this end, we employ a graph-based shape description and matching methods to return possible apparel shape solutions proactively.

5.1. The Criteria for Recommendation Selection. In order to determine which apparel shapes are chosen from the database for recommendation, a partial matching algorithm is employed. The shapes with the highest similarities to the input drawing are selected and returned to the designer in real time once a new command input is detected.

There are generally three cases. Firstly, if a candidate garment shape includes the incomplete input shape, they are considered to be highly similar since the input shape can be finished later. Secondly, if the partial input shape includes certain components that are not included in the candidate shape or includes more components than the candidate, the candidate shape is regarded not to conform to the user requirement, and the similarity should be very low even if other corresponding parts may be very similar. Thirdly, if the partial input shape is included by two or more candidate garment shapes, the one with the fewest components will have the highest similarity.

5.2. Dynamic Matching Procedure. There are mainly two modules involved in our intelligent apparel recommendation procedure, including garment shape description and partial matching.

In the shape description module, both geometrical and topological features of the garment shapes are extracted. A bisegment descriptor proposed in our previous work [3, 12, 13] is used to model these shape features into graph structures. In this way, the raw sketch input data is transformed into graph representations, which are then used for the following

partial matching. We introduce the calculation of the bisegment graph (BSG) shape descriptor in the following.

Definition 5 (bisegment graph). A BSG is defined as a 6-tuple $G = (V, E, R_E, \Theta_E, \Lambda_E, C_E)$, where

(i) V is the set of graph nodes (every node corresponds to a segment in the garment shape),

(ii) $E \subseteq V \times V$ is the set of edges of the graph (if there is a bisegment combining two neighbor segments v_i and v_j, an edge $e_{ij} = (v_i, v_j)$ exists),

(iii) $R_E : E \rightarrow \Sigma_R$ is a function assigning the feature of binary topology relations $\Sigma_R = \{R_{l,l}, R_{l,c}, R_{c,c}\}$ to the edges,

(iv) $\Theta_E : E \rightarrow (0, \pi]$ is a function extracting the geometrical features by calculating the inner angle of the two adjacent segments of a bisegment,

(v) $\Lambda_E : E \rightarrow (0, 1]$ is a function extracting the geometrical features by calculating the ratio between the segment lengths of the bisegment,

(vi) $C_E : E \rightarrow [0, \pi) \times [0, \pi)$ and $C_E(e_{ij}) = (\kappa_1, \kappa_2)$, where κ_1 and κ_2 are the curvatures of the two segments at their intersection point (here κ_1 and κ_2 are sorted in an ascending order to count for rotation invariance).

Figure 4 shows the procedure of bisegment graph calculation. The original shape is first regularized and segmented into $\{a, b, c, d\}$ before they are decomposed into a bisegment sequence as $\{B_1, B_2, B_3, B_4\}$. The features are then extracted, and a bisegment graph is constructed based on Definition 5.

In the partial matching module, a partial matching algorithm is developed to calculate the similarity for dynamic apparel shape recommendation. The partial matching of garment shapes is performed between the precollected shapes in the apparel database and the user's input shape from the hypertouch interface. The calculation of the similarity is asymmetric. We carry out the partial matching based on a product graph spanning algorithm [13]. A weighted product graph of two bisegment graph models representing the difference between two garment shapes is derived. The similarity is then calculated from the minimum spanning tree (MST) of the weighted product graph.

The process of our intelligent apparel recommendation is shown in Figure 5, which gives an example of the matching procedure between two garment shapes. Figure 5(a) is a candidate garment shape in the database. Its corresponding bisegment graph representation is shown in Figure 5(b). Figure 5(c) is the incomplete sketch shape input drawn by the designer. The bisegment feature of the input is calculated online to generate the bisegment graph descriptor in Figure 5(d). The product graph of the bisegment graphs for the candidate shape and input shape is derived from the Kronecker product operation [13], and the result is given in Figure 5(e). We will explain the details of the dynamic partial matching algorithm in the following.

5.2.1. Product Graph Construction. We will first define some basic concepts.

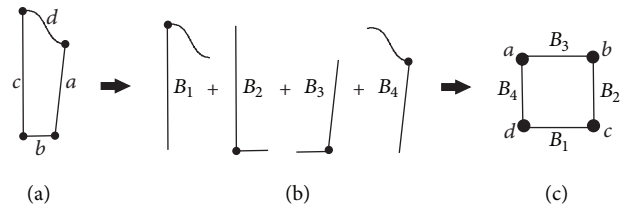


Figure 4: The calculation of the bisegment graph of a garment shape. (a) A regularized shape for the sleeve part of a garment. (b) The decomposition of the bisegments. (c) The graph construction of the corresponding BSG model.

Definition 6 (modified Kronecker product). Given two real matrices, $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{r\times s}$, the modified Kronecker product $A \overline{\otimes} B \in \mathbb{R}^{mr\times ns}$ is defined as

$$A \overline{\otimes} B = \begin{pmatrix} abs\left(\left[A_{11}\right]_{r\times s} - B\right) & \dots & abs\left(\left[A_{1n}\right]_{r\times s} - B\right) \\ \vdots & \ddots & \vdots \\ abs\left(\left[A_{m1}\right]_{r\times s} - B\right) & \cdots & abs\left(\left[A_{mn}\right]_{r\times s} - B\right) \end{pmatrix}, \tag{4}$$

where $[A_{ij}]_{r\times s}$ denotes a $r \times s$ real matrix with elements of A_{ij}, and $abs(A)$ is a $r \times s$ matrix with elements of $abs(A)$.

The product of two weighted graphs can then be defined based on Definition 6.

Definition 7 (weighted direct product graph (WDPG)). Given two graphs, $G_1(V_1, E_1)$, $G_2(V_2, E_2)$, and their weighted matrices as $W_1 \in \mathbb{R}^{m\times n}$ and $W_2 \in \mathbb{R}^{r\times s}$, respectively, their weighted direct product graph $G_{1\times 2}$ is a graph with vertex set $V_{1\times 2} = \{(u_i, v_r) : u_i \in V_1, v_r \in V_2\}$, edge set $E_{1\times 2} = \{((u_i, v_r), (u_j, v_s)) : (u_i, u_j) \in E_1 \wedge (v_r, v_s) \in E_2\}$, and weighted matrix $W_{1\times 2} = W_1 \overline{\otimes} W_2$.

From Definition 7, the WDPG $G_{1\times 2}$ is a product graph that each vertex involves a vertex pair over two graphs G_1 and G_2, an edge indicates that the corresponding vertices in the original graphs are both neighbors, and the weights of the edges are determined by $W_{1\times 2} = W_1 \overline{\otimes} W_2$ from Definition 6.

The construction of WDPG of two graphs of different sizes is shown in Figure 5. The vertex in the product graph corresponds to a vertex pair in the original graphs, and an edge exists if and only if the corresponding vertices are neighbors in both original graphs. The weights of the product graph are set by calculating the absolute difference of the weights of the corresponding edges from the original graphs. For instance, the weight of edge $(A1, B2)$ in the product graph is set to be 3, which is the absolute difference value of the weights of edge (A, B) and $(1, 2)$ in the original graphs.

5.2.2. Similarity Measurement. Here we give the matching metric based on the MST.

For a product graph, given its minimal spanning tree with edge set denoted as $\{e_{i_1}, e_{i_2}, \dots, e_{i_{|V|-1}}\}$ and the weight set

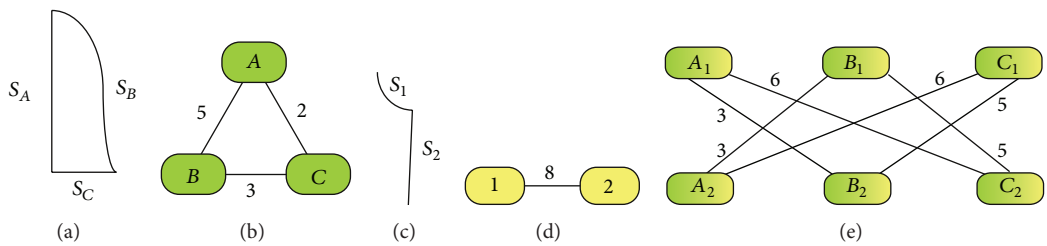


Figure 5: The partial matching procedure. (a) A candidate garment shape with three segments. (b) The bisegment graph extraction of (a). (c) An incomplete garment shape from the designer with two segments. (d) The bisegment graph extraction of (b). (e) The weighted directed product graph of (b) and (d) by a modified Kronecker product operation.

as $\{w_{i_1}, w_{i_2}, \ldots, w_{i_{|V|-1}}\}$, then the proposed similarity metric between two original graphs is determined by

$$d(G_1, G_2) = \sum_{k=1}^{|V|-1} \frac{w_{i_k}}{|V|-1}. \quad (5)$$

According to our calculation of the weights of the product graph, which encodes the difference of weights between two corresponding edges in two original graphs, the edge-matching cost is captured by the minimal spanning tree of the product graph. In this way, the above measurement based on minimal spanning tree can capture the minimal matching cost between two original graphs.

In this paper, the Prim algorithm [14] is used to implement the minimal spanning tree of the product graph. The time complexity is $O(nm\ \log(n)+|E|)$, where n and m $(m \leq n)$ denote the sizes of the original graphs and $|E|$ denotes the number of edges of the product graph (here, given two BSG G and G' of size n and m, the size of their product graph is nm).

Also note that since the product graph can be constructed between graphs of different sizes, the proposed matching metric is capable of both complete and partial matching.

6. Performance Evaluation

In order to evaluate the efficiency and the effectiveness of our proposed method, we develop a prototype system of our intelligent recommendation for hypertouch apparel design. The setup of our experiment is a workstation of Windows 7 with an Intel Core i7 2.8 GHz CPU, 8 GB RAM, and GeForce GTX 1792 MB. The environment of hypertouch interaction is built upon a large freehand tabletop surface [15] which offers dynamic finger tracking and gesture recognition.

First, the garment shape database in our previous work [3, 12] is extended. 10 garment designers are invited to create standard garment shapes using CAD software, and we achieve a new shape database of 454 garment shapes. The database is categorized based on the standard panel models. Given an input shape, the retrieved targets are considered to be successful if they fall into the same category as the input shape. We compare the proposed method with the state-of-the-art garment shape matching methods [3, 12, 16]. The demonstration of our hypertouch interface and the result of intelligent apparel recommendation by dynamic matching is given in the following.

6.1. Gesture Interaction for Hypertouch Garment Design. The designing process is described as follows:

(1) the designer starts off with a virtual mannequin;

(2) the designer uses the hypertouch interface to carry out manipulations in the hypertouch environment;

(3) designers draw an outline of the desired clothing over the mannequin;

(4) our system converts the outline into incomplete clothes shapes and searches for similar garment shapes in the database;

(5) garments that fit most of the input shapes are returned to the user dynamically along with the drawing process.

Figure 6 gives a demonstration of our hypertouch gesture interaction for interactive apparel shape design. The comments from designers show that our interface helps to have fast prototyping of apparel shapes and garment design. From Figure 6 we can see that the apparel modeling system is designed to have intelligent computer supports, which still offers a similar feeling to real life design on a mannequin. Flexible and quick controls of the garment shape are enabled to accomplish the design task quickly.

6.2. Results for Intelligent Apparel Recommendation. Figure 7 shows the dynamic recommendation results given a front and a skirt shape input. As we can see, the user intention of the desired shape is predicted along with the completion of the shape drawing. Recommendations of the garment shape are returned dynamically once a user input is detected. In this way, the apparel design process is largely speeded up by our intelligent recommendation.

We then compare the performance of the proposed dynamic panel matching algorithm with other matching methods. We calculate the similarity between the input shapes and the garment shapes in the database. The top-K $(K = 5, 10, 15, 20)$ garment shapes are returned as the matched results. All results are averaged over all input trials. The same shape features are used for all comparing algorithms.

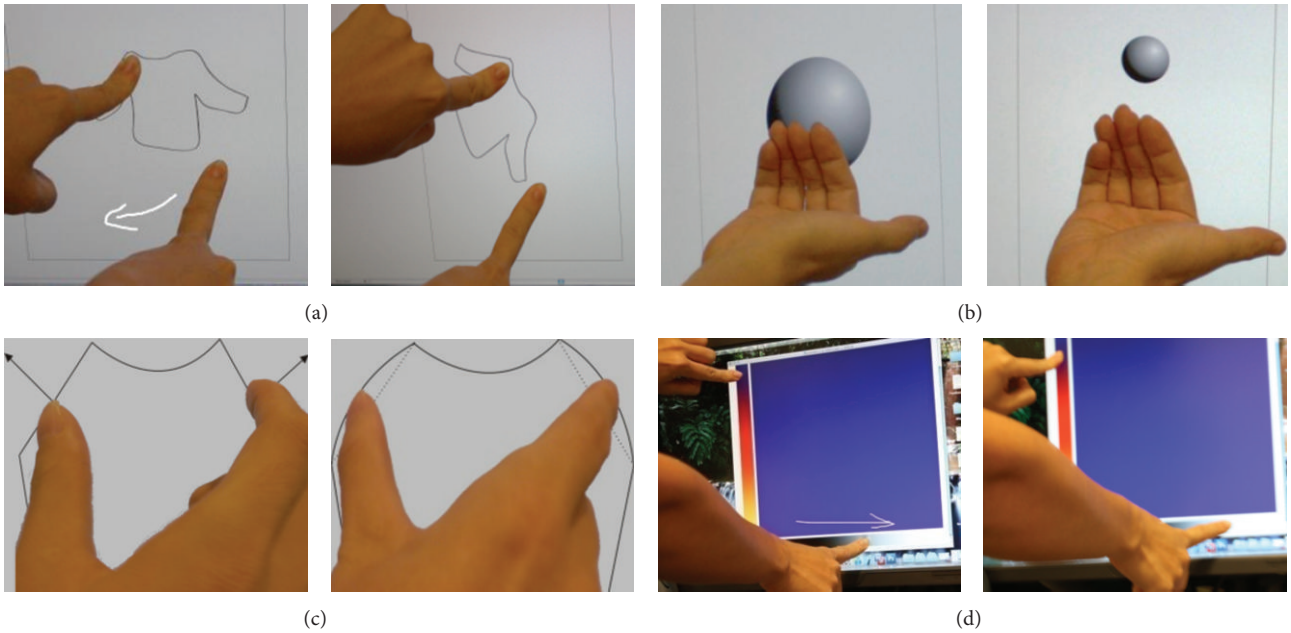


Figure 6: Demonstrations of hypertouch gestures for apparel design. (a) A rotation gesture performed by one hand pressing and holding on the object and the other hand specifying the direction of the movement. (b) A 2.5D zoom-in gesture by waving the object closer to the user. (c) A shape blending gesture by two fingers moving apart. (d) A color control gesture by one hand selecting the color and the other hand adjusting the gradient degree.

The well-known precision and recall rates are used to evaluate the matching performance of the similarity metrics. With the number of returning items increasing, the precision will be decreased but the recall will be increased. Recall-precision curves can be plotted based on the recall and precision pairs, in which a higher curve indicates better matching performance. The response time of the matching procedure is also measured. We compare our proposed matching method with the state-of-the-art garment shape matching methods, including user behaviour tree (UBT) matching [3, 12] and attribute strings matching [16].

The matching accuracy with K returned items is given in Figure 8. It is shown that the proposed matching approaches achieve the best performance among all methods. The average matching precision even achieves 0.95 with $K = 5$, which shows that our metric is effective to calculate the similarity between two garment shapes.

The recall-precision curves are shown in Figure 9. Same as above, the proposed method achieves the highest performance among all methods. This further confirms that our proposed method can reach better performance than the state-of-the-art methods.

We also calculate the average response time of the proposed approach, which is about 50 ms. Moreover, for a novel user, the whole garment design process takes around 2 minutes including drawing, editing, correcting, and selecting candidate garments. This is sufficiently fast for real time garment design task.

7. Conclusions

In this paper, we propose an intelligent apparel design system that supports dynamic recommendation, or the so-called recommendation in motion, to capture user preferences online. Prior domain knowledge is extracted, and a database of user drawing patterns is constructed accordingly. A dynamic partial matching algorithm based on minimal spanning tree is used to calculate the similarity between the input shape and the candidate shapes in the apparel database. Possible shape solutions are ranked in the order of similarity descending and returned to the user proactively with the shape creation progress. In this way, the efficiency of apparel design is largely improved. Meanwhile, the designer and the computer system are connected and communicated in a more intelligent way than with transitional CAD software.

We also present a new concept of hypertouch interaction. Higher degrees of freedom and more information are achieved in the hypertouch environment. Nonrigid sketching and gesture interaction are also supported with hypertouch features. This benefits the touch-based interface towards a more flexible, intuitive, efficient, and easy-to-use user experience.

One direction of our future work is the development of 3D hypertouch interface. How to robustly track the hands and recognize the gestures in 3D still attracts lots of attentions. Considering Kinect or LeapMotion instead of tabletop table will be a possible solution. Another direction is to increase the volume of the apparel database to accommodate a larger variety of garment shapes.

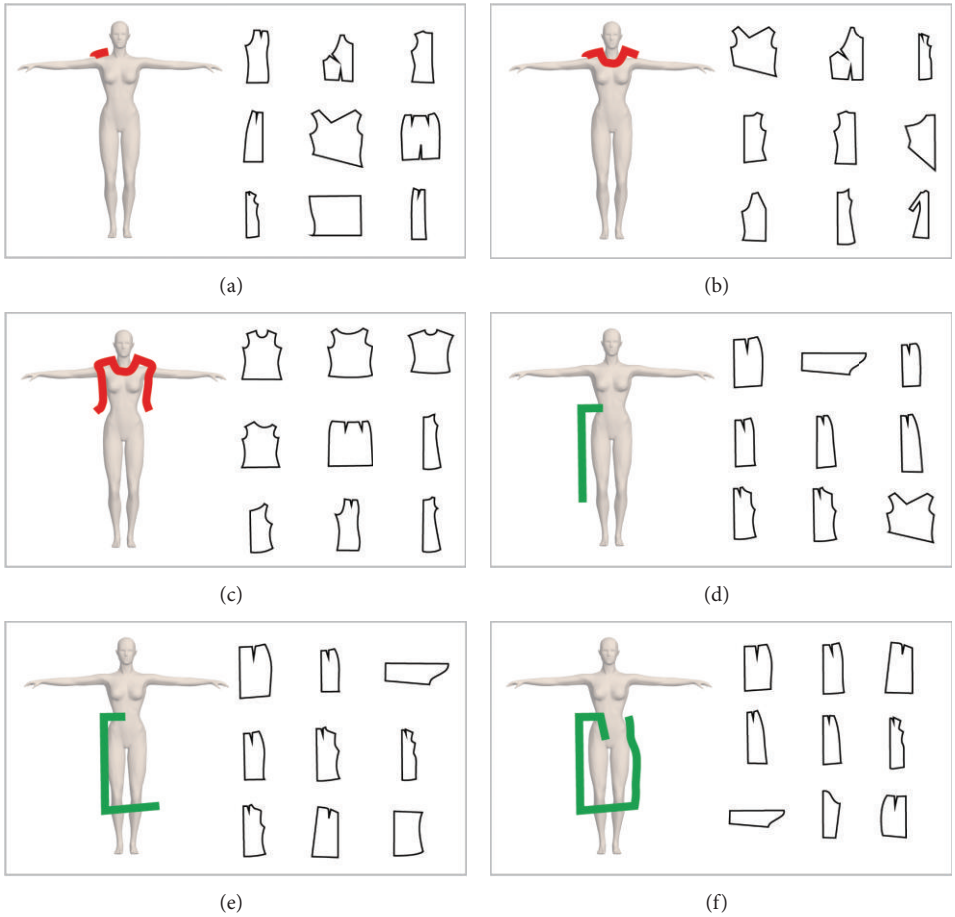


Figure 7: Dynamic recommendation for the apparel design. The recommendation results are shown on the right side of the design panel, with similarities descending from left to right and top to bottom. (a)–(c) The intelligent designing process of a front garment shape by one user. (d)–(f) The intelligent designing process of a skirt garment shape by another user.

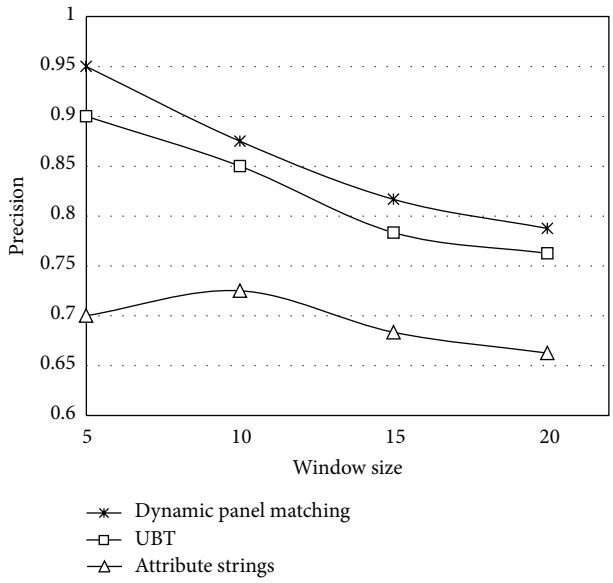


Figure 8: Average precision rates of different return window sizes.

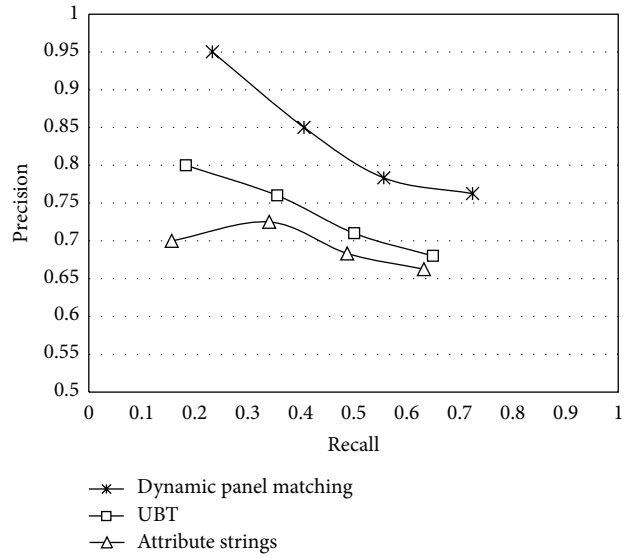


Figure 9: The precision-recall curves of different panel matching methods.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported by The Hong Kong Research Grants Council General Research Fund Grants (nos. PolyU 5101/09E, PolyU 5101/08E), The National Science Foundation of China (nos. 61272276, 61305091), The National Twelfth Five-Year Plan Major Science and Technology Project (no. 2012BAC11B00-04-03), The Fundamental Research Funds for the Central Universities (no. 2100219038), and Shanghai Pujiang Program (no. 13PJ1408200). The authors would also like to thank the anonymous reviewers for their valuable comments and suggestions.

References

[1] C. K. Au and Y.-S. Ma, "Garment pattern definition, development and application with associative feature approach," *Computers in Industry*, vol. 61, no. 6, pp. 524–531, 2010.

[2] S. K. Lee, W. Buxton, and K. C. Smith, "A multi-touch three dimensional touch-sensitive tablet," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '85)*, pp. 21–25, ACM, New York, NY, USA, 1985.

[3] S. Liang, E. C. L. Chan, G. Baciu, and R.-H. Li, "Cognitive garment design interface using user behavior tree model," in *Proceedings of the 9th IEEE International Conference on Cognitive Informatics (ICCI '10)*, pp. 496–500, Beijing, China, July 2010.

[4] D. Wigdor, J. Fletcher, and G. Morrison, "Designing user interfaces for multi-touch and gesture devices," in *Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems (CHI '09)*, pp. 2755–2758, ACM, New York, NY, USA, April 2009.

[5] H. Benko, A. D. Wilson, and P. Baudisch, "Precise selection techniques for multi-touch screens," in *Proceedings of the Conference on Human Factors in Computing Systems (CHI '06)*, pp. 1263–1272, ACM, New York, NY, USA, April 2006.

[6] M. Wu and R. Balakrishnan, "Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays," in *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology (UIST '03)*, pp. 193–202, ACM, New York, NY, USA, November 2003.

[7] K. Hashiya and S. Kojima, "Acquisition of auditorycvisual intermodalmatching-to-sample by a chimpanzee (pan troglodytes): comparison withvisual visual intramodal matching," *Animal Cognition*, vol. 4, pp. 231–239, 2001.

[8] D. Freeman, H. Benko, M. R. Morris, and D. Wigdor, "ShadowGuides: visualizations for in-situ learning of multi-touch and whole-hand gestures," in *Proceedings of the 4th ACM International Conference on Interactive Tabletops and Surfaces (ITS '09)*, pp. 165–172, ACM, New York, NY, USA, November 2009.

[9] J. O. Wobbrock, M. R. Morris, and A. D. Wilson, "User-defined gestures for surface computing," in *Proceedings of the 27th International Conference on Human Factors in Computing Systems, ser. (CHI '09)*, pp. 1083–1092, ACM, New York, NY, USA, 2009.

[10] T. Igarashi and J. F. Hughes, "Clothing manipulation," in *Proceedings of the 15th Annual Symposium on User Interface Software and Technology*, pp. 91–100, ACM, New York, NY, USA, October 2002.

[11] E. Turquin, J. Wither, L. Boissieux, M.-P. Cani, and J. F. Hughes, "A sketch-based interface for clothing virtual characters," *IEEE Computer Graphics and Applications*, vol. 27, no. 1, pp. 72–81, 2007.

[12] S. Liang, R.-H. Li, G. Baciu, E. C. L. Chan, and D. Zheng, "Partial matching of garment panel shapes with dynamic sketching design," in *Proceedings of the 1st Augmented Human International Conference (AH '10)*, pp. 1–6, ACM, New York, NY, USA, April 2010.

[13] S. Liang, R.-H. Li, and G. Baciu, "A graph modeling and matching method for sketch-based garment panel design," in *Proceedings of the 10th IEEE International Conference on Cognitive Informatics and Cognitive Computing (ICCI '11)*, pp. 340–347, August 2011.

[14] C. E. L. Thomas, H. Cormen, and R. L. Rive, *Introduction to Algorithms*, The MIT Press, Cambridge, Mass, USA, 3rd edition, 2009.

[15] PQLabs, "3d engine," 2009, http://multi-touch-screen.com/support.html.

[16] E. Ozcan and C. K. Mohan, "Partial shape matching using genetic algorithms," *Pattern Recognition Letters*, vol. 18, pp. 987–992, 1997.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 549838, 21 pages
http://dx.doi.org/10.1155/2013/549838

Research Article

Design of an Automatically Tuned Fuzzy Controller for a Truck and Multitrailer System

T. R. Ren

School of Mechanical and Manufacturing Engineering, The University of New South Wales, Sydney 2052, Australia

Correspondence should be addressed to T. R. Ren; trren@hotmail.com

Received 9 April 2013; Revised 19 June 2013; Accepted 20 June 2013

Academic Editor: Shengyong Chen

The use of a truck and multitrailer system is advantageous because of its ability to transport heavy and large parts with a single powered vehicle. On the other hand, when the system is deployed in an autonomous and unmanned scenario, it remains a challenging task to design a drive controller. Since the drive is only applied to the truck and motivated by successful cases of human expert drivers, a fuzzy controller is developed to generate speed and turn rate commands in order to steer the multi-trailer system to reach the target position with minimum position error. Furthermore, in order to make the controller design efficient and effective, the parameters in the fuzzy controller including the membership functions and rules are automatically tuned using the implementation of efficient particle swarm optimization algorithm instead of relying solely on human expert knowledge. Near-optimal parameters are then derived and adopted in the controller, and drive commands are then generated. The performance of the truck-and-multi-trailer system under fuzzy control is verified through simulation studies, and satisfactory results are obtained.

1. Introduction

There has been a huge demand for the transportation of manufactured products from the plant to destinations for storage or dispatch. When the item to be carried is extremely large, for example, the components of an airplane [1], the use of a truck and multitrailer (TMT) would be advantageous against using a single vehicle. On the other hand, it is very difficult in driving such system due to the fact that only the truck can be manipulated while the trailers are passive. Skillful drivers are therefore always needed [2], but it is also desirable that automatic driving can be deployed while manipulation difficulties can be mitigated as much as possible [3].

The difficulties that hinder the operation of the truck system may be attributed to the complicated configuration of the trailer chain. In general, the trailer structure can be grouped according to the location of joints into the off-hooked, direct-hooked, and three-point structures. Trade-offs are required for mechanical simplicity and control limitations. It has been shown that the off-hooked configuration is able to provide a smaller steady state tracking error [4]. Moreover, based on the properties of the trailer system assembly [5], the steering of the truck has to be limited [6] in order to avoid collisions among the truck and the trailers. While attempting to achieve satisfactory control performance, it has also been shown that the problematic jackknife phenomenon can be avoided with dynamic output feedback with a reduction in the use of joint sensors [7]. Among the available control methods, nonlinear control approaches based on Lyapunov and back-stepping designs had been employed [8]. These methods are advantageous in tackling problems arising from the control of articulated and underactuated systems. In a similar context, the feedback stabilization scheme was adopted in controlling and stabilizing a truck-trailer system [9].

One of the common scenarios that a truck-trailer system employs for transporting parts is to follow a desired trajectory from the start to the final position [10]. As it can be seen, the control of the truck-trailer system to follow a desired path is nontrivial [11] due to nonholonomic constraints and limits in steering angles. This problem may become more severe if there are uncertainties in measuring the joint

angles between trailers [12]. Furthermore, the effect of trailer configurations had been investigated for on-axle and off-axle hitching modes [13]. The feasibility of employing Lyapunov and back-stepping design was demonstrated. In the work reported in [14], a virtual articulated vehicle was inserted into the trailer chain to facilitate the control for reverse trajectory tracking. It has also been highlighted [15] that the trajectory curvature also plays an important role in the tracking performance of a truck-trailer system. Moreover, an alternative approach based on geometry features of the truck-trailer kinematics was proposed [16]. The method considered the cascaded kinematics and the propagation of the drive command through the system.

While model-based control theoretic approaches are able to derive controllers for the truck-trailer system with satisfactory performance, their design could be severely affected by the uncertainty in the model and unpredictable disturbances while driving the truck-trailer system. For these cases, soft computing techniques such as fuzzy theories may be useful in the design of mobile robots and truck-trailer system controllers [17, 18]. In the fuzzy controller, a rule base is formulated, and with the use of membership functions describing the input and output variables, an inference engine is then invoked to generate the desired output. In [19], the design of a fuzzy controller was accompanied with a fuzzy observer for better performance. Further examples of employing fuzzy control for truck-trailer systems had also been presented in [20–22] which had incorporated the use of Lyapunov design, line-of-sight method, and constraints imposed from obstacle avoidance.

To a large extent, the design of fuzzy controllers originated and relied on the availability of expert knowledge of the behavior of the system under consideration. Hence, the design of controllers may be time consuming and tedious. Therefore, techniques were developed for setting fuzzy rules by learning from test examples [23]. This approach had been well received, and the incorporation of neural network based methods was reported in [24] with the aim to facilitate training the fuzzy controller. Further to effects laid in releasing the burdens in building the fuzzy system, the use of intelligent [25] and evolutionary computing methods is also a very attractive approach [26]. These attempts include the genetic algorithm [27] which was inspired by the process in which living species evolve according to the principle of the survival of the fittest. Due to its generality, the algorithm is able to tune the fuzzy controller parameters to near-optimal values. Recently, the quantum-inspired evolutionary algorithm has been used to tune the parameters of a fuzzy controller [28]. This algorithm possesses a wider diversity of potential solutions which may be advantageous in searching the optima.

Though these evolutionary and intelligent computing based methods produce fuzzy controllers for truck-trailer systems with satisfactory performances, their implementation may not be efficient. It is mainly because the algorithmic complexity and overheads are imposed in addition to problem dependent evaluation of objective functions. To this end, the particle swarm optimization algorithm (PSO) [29, 30] could be employed in tuning fuzzy controllers for its implementation simplicity. The use of the PSO algorithm, inspired from mimicking swarms of agents moving through the solution space, has been very attractive because of its generic applicability to a wide domain of problems. For example, it was used in the coordination of a swarm of mobile robots [31], robot motion planning [32], and in the design of fuzzy controllers [33]. Additional examples on the use of PSO in fuzzy controller designs can be found in [34, 35]. The fuzzy membership functions and fuzzy rules were tuned using PSO, and satisfactory results had been obtained in the control of industrial processes.

In this work, a fuzzy controller is designed for a truck-trailer system to reach a desired position. In order to be released from depending solely on human expert knowledge in the controller design an intelligent computation algorithm, the particle swarm optimization, is adopted for its implementation efficiency. In particular, fuzzy membership function coefficients and fuzzy rules are tuned simultaneously using the PSO algorithm. The generality of the proposed approach is demonstrated when the method is used to tune fuzzy controllers for truck-trailer systems with single and multiple trailers. The effectiveness of the developed controller is illustrated using simulation studies.

The rest of this paper is organized as follows. In Section 2, the kinematic model of the truck-trailer system is given. The fuzzy controller design is detailed in Section 3. Simulations are described in Section 4 with results presented and discussed. Finally, a conclusion is drawn in Section 5.

2. System Model

A system model is developed here describing the kinematic relationships between the driving truck and the passive trailers. The positions of the centers of the trailers are derived from the position of the truck. The translation and angular velocities of the trailers are then formulated with respect to the drive commands applied at the truck. Finally, the construction of the overall truck-and-multi-trailer system is presented.

2.1. Positional Model. Let the truck-and-multi-trailer system consist of an actively driven truck and a number of passible trailers. The truck and trailers, collectively named as vehicles, are connected at joints along the front-to-end axis of each vehicle [4, 5]. Figure 1 shows a diagram of the typical connection between two vehicles i and $i+1$.

The truck-and-multi-trailer system is deployed in a flat terrain described by a world coordinate xoy. The center of the wheels is taken as the vehicle position labeled as (x_i, y_i), and the vehicle is inclined to an angle θ_i with respect to the world coordinate x-axis. The size of the vehicle is denoted as the length F_i from the wheel center to the front and R_i to the rear. The angular difference between two consecutive vehicles is given by $\beta_{i+1} = \theta_i - \theta_{i+1}$. The indices increase towards the last trailer.

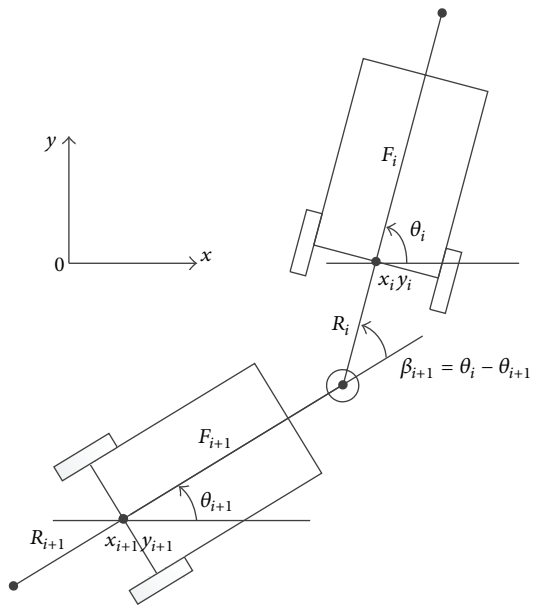


Figure 1: Truck-and-multi-trailer system consisting of connected vehicles.

By making use of the definitions of the parameters of the TMT system, the general positions of the vehicles can be determined from

$$x_{i+1} = x_i - R_i \cos\theta_i - F_{i+1} \cos\theta_{i+1},$$
$$y_{i+1} = y_i - R_i \sin\theta_i - F_{i+1} \sin\theta_{i+1}. \quad (1)$$

2.2. Kinematic Model. For the vehicles inclination angles θ_i, they are determined by considering the effect of the driving efforts applied to the leading vehicle. Let the ith vehicle be driven to a translational velocity v_i and an angular velocity $\dot{\theta}_i = \omega_i$; see Figure 2.

2.2.1. Translational Velocity. The resolved translational velocity along the front-to-end axis of vehicle $i + 1$ due to the translational velocity of the leading vehicle i is

$$v_i \cos\beta_{i+1}. \quad (2)$$

Moreover, the translational velocity component; arising from the angular velocity applied to vehicle i and effected on $i + 1$th vehicle, is first obtained from the rotation of the R_i link, giving $R_i\dot{\theta}_i$, It is then resolved along the direction of the front-to-end axis of the $i + 1$th vehicle as

$$R_i \sin\theta_{i+1}\dot{\theta}_i. \quad (3)$$

Hence, the total translational velocity of the $i + 1$th vehicle becomes

$$v_{i+1} = v_i \cos\beta_{i+1} + R_i \sin\theta_{i+1}\dot{\theta}_i. \quad (4)$$

2.2.2. Angular Velocity. For the angular velocity of the $i + 1$th vehicle, due to the drive from the ith vehicle, is obtained by balancing the rotations (Figure 3).

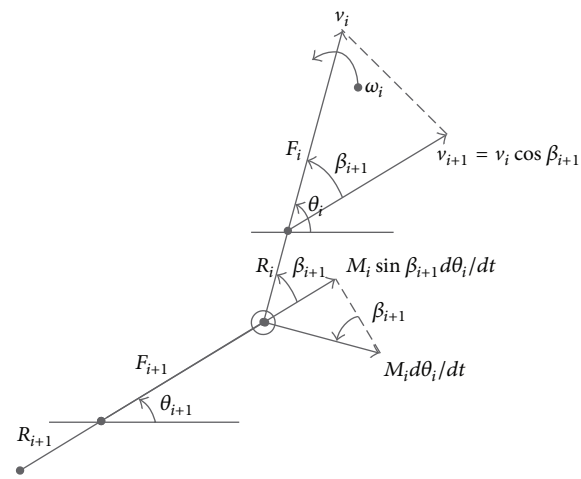


Figure 2: Propagation of translational velocity.

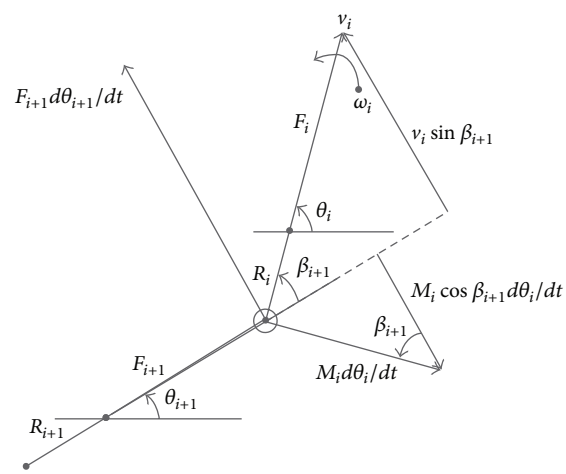


Figure 3: Propagation of angular velocity.

The component from the rotation of $i + 1$th vehicle is

$$F_{i+1}\dot{\theta}_{i+1}. \quad (5)$$

The element arising from the translational velocity of the ith vehicle is

$$v_i \sin\beta_{i+1}. \quad (6)$$

Furthermore, the portion of the angular velocity due to the link R_i is obtained from the effect caused by the rotation of the ith vehicle and then resolved along the perpendicular to the front-to-end axis of the $i + 1$th vehicle; that is

$$R_i \cos\beta_{i+1}\dot{\theta}_i. \quad (7)$$

Hence, the resultant angular velocity of the $i + 1$th vehicle is

$$\dot{\theta}_{i+1} = \frac{1}{F_{i+1}}\left(v_i \sin\beta_{i+1} - R_i \cos\beta_{i+1}\dot{\theta}_i\right). \quad (8)$$

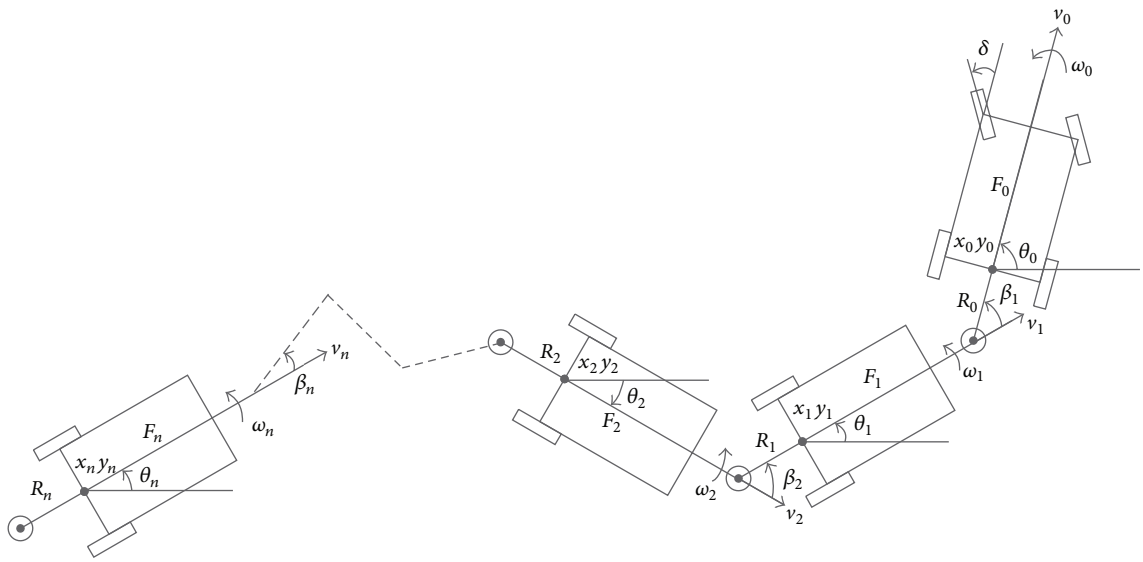


Figure 4: Truck-and-multi-trailer system with n trailers.

2.3. Motion of the Truck and Multitrailer System. Now (4) and (8) have provided the translational and angular velocities of a generic vehicle with regard to the velocities of a leading vehicle in the TMT system. Also note that the truck is the head of the truck-trailer chain, and its motion is driven by given commands as its translational and angular velocities, namely, v_0 and $\dot{\theta}_0 = \omega_0$ (Figure 4). Then by iterating through the TMT system chain, the velocities can be found. Based on the translational and angular velocities applied to the truck which is equipped with a steering wheel, the angle sustained with respect to the truck front-to-end axis is

$$\delta = \tan^{-1}\left(\frac{\omega_0 F_0}{v_0}\right). \tag{9}$$

Furthermore, by invoking (1), the positions of the centers of the vehicles can be found while the dimensions of the links F_i and R_i had been given as system parameters, and the initial intervehicle angles β_i are also known. We have

$$\begin{aligned} v_1 &= v_0 \cos\beta_1 + R_0 \sin\beta_i \omega_0, \\ \dot{\theta}_1 &= \frac{1}{F_1}\left(v_0 \sin\beta_1 - R_0 \cos\beta_1 \omega_0\right). \end{aligned} \tag{10}$$

In addition,

$$\begin{aligned} v_2 &= v_1 \cos\beta_2 + R_2 \sin\beta_2 \dot{\theta}_1, \\ \dot{\theta}_2 &= \frac{1}{F_2}\left(v_1 \sin\beta_2 - R_1 \cos\beta_2 \dot{\theta}_1\right). \end{aligned} \tag{11}$$

The velocities of other trailers in the system can then be obtained iteratively by referencing to an ordered indexing convention. Together with the rate of change of vehicle orientation,

$$\dot{\beta}_2 = \dot{\theta}_1 - \dot{\theta}_2, \tag{12}$$

the positions and orientations of the vehicles can be determined.

3. Controller Design

In order to steer the truck-and-multi-trailer system to a desired target position, a controller is designed to generate the drive commands. In particular, to simplify the design, a direct-hooked model is used. Further, to cater for the nonlinearity and complexity, a fuzzy logic controller is also proposed.

3.1. Control Strategy. The desired motion is obtained by driving the pair of truck wheels with a commanded linear velocity v and turn rate ω. Let the system state be parameterized by

$$\mathbf{q} = \left[x_0, y_0, \theta_0, \beta_1, \ldots, \beta_{n-1}\right]^T, \quad \mathbf{q} \in \mathbb{R}^{n+3}, \tag{13}$$

where x_0, y_0, and θ_0 are the position and orientation of the truck with respect to a world coordinate, β_i is the angle of the vehicle with a vehicle in front, and n is the number of trailers. It is also assumed that there is no slip on any of the wheels.

There are three possible configurations of truck-trailer systems, which are off-hooked, direct-hooked, and three-point trailer systems. From the previous work [4], the mechanical structure of three-point trailer system is very complicated. Direct-hooked trailer system is a special condition of off-hooked trailer system. In such scenario, the length of the rear link becomes diminished; that is, $R_i = 0$.

In order to avoid the jackknife problem, the angle β_i is limited to within

$$-\frac{\pi}{2} \le \beta_i \le \frac{\pi}{2}, \tag{14}$$

```
for each particle do
    Initialize the particle positions and their velocities randomly
end for
Set a finite number of iterations
Set iteration count to 0
while not terminate do
    for each particle do
        Evaluate the fitness value
        if the fitness value is better than the local best fitness value (pBest) in history then
            Update the local bests and their fitness
        end if
    end for
    Choose the particle with the best fitness value of all the particles as the global best (gBest)
    for each particle do
        Calculate particle velocity
        Update particle position
    end for
end while
Terminate if generation count expires
```

Algorithm 1: The PSO algorithm.

where $i = 1, 2, \ldots, n$. The range of turning rate in rad/s is

$$-\frac{\pi}{3} \leq \omega \leq \frac{\pi}{3}. \tag{15}$$

Furthermore, the positional errors are given by

$$\begin{aligned} E_x &= x_e - x_0(t), \\ E_y &= y_e - y_0(t), \\ E_\theta &= \theta_e - \theta_0(t), \end{aligned} \tag{16}$$

where x_e, y_e, and θ_e define the desired ending system position and E_x, E_y, and E_θ represent the distance error in the direction parallel to the x- and y-axes and the desired truck orientation. The tracking errors E_{total} are calculated by

$$E_{\text{total}} = \sqrt{E_x^2 + E_y^2 + E_\theta^2}. \tag{17}$$

A fuzzy logic controller is able to transfer uncertain experience and knowledge about the behaviour of observed objects into the process of decision making based on fuzzy rules and membership functions. Therefore, how to generate both rules and membership functions plays a very important role in the fuzzy controller design. If the exact knowledge about the dynamics or behaviour of the observed system is available, the rule and membership function generations are straightforward. However, if knowledge and experience are limited, the controller design task will be very difficult. In this section, the generation of fuzzy logic controller rules and membership functions is formulated as an optimization problem, whereby the objective is to specify the controller parameters to the satisfaction of all constraints. The particle swarm optimization algorithm is employed to solve this optimization problem.

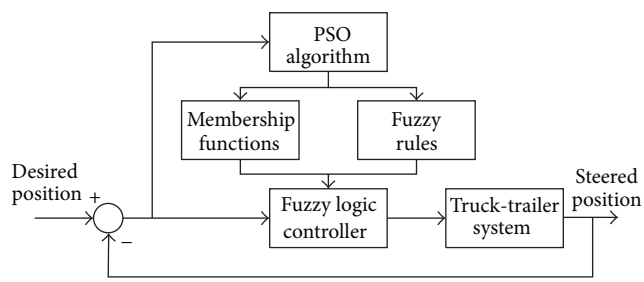
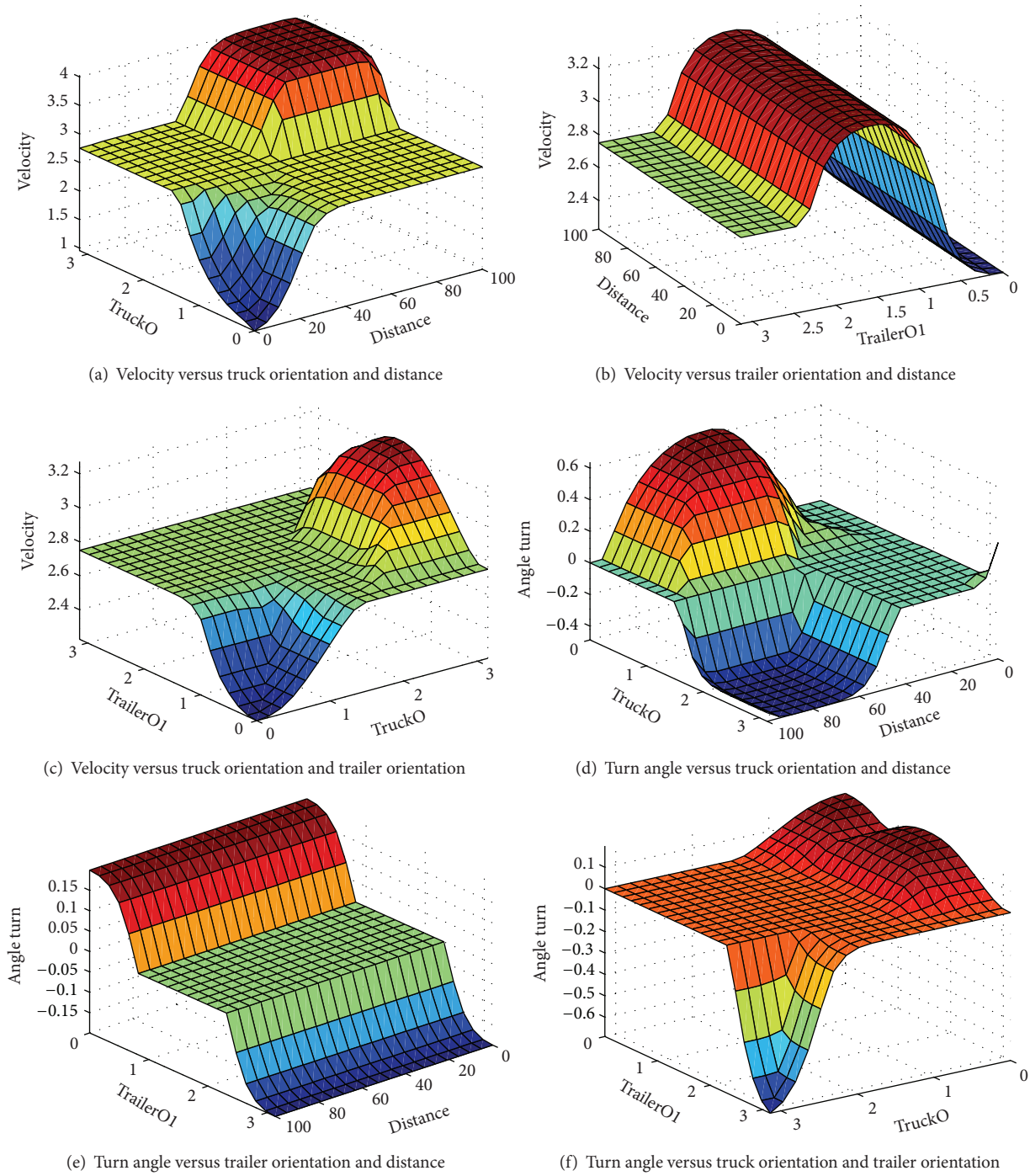


Figure 5: The proposed fuzzy logical controller based on PSO.

3.2. Particle Swarm Optimization. The particle swarm optimization algorithm belongs to an agent-based heuristic search method used to find the maxima/minima of an objective function in a search space, whose potential solutions are coded as particles. The algorithm contains a recursive loop of iterations and could be described by Algorithm 1. In addition, the PSO algorithm can be described by the following expressions:

$$\begin{aligned} v_{p,k+1} &= wv_{p,k} + c_1\left(z_{g,k} - z_{p,k}\right) + c_2\left(\widehat{z}_{p,k} - z_{p,k}\right), \\ z_{p,k+1} &= z_{p,k} + v_{p,k+1}, \end{aligned} \tag{18}$$

where w is a weighting factor, v is the velocity of the practical in the solution space. c_1 and c_2 are random numbers in $U[0, c_{1,\max}]$, $U[0, c_{2,\max}]$; here, $U[\cdot,\cdot]$ presents the uniform distribution and $c_{1,\max}$ is the maximum value for c_1 and $c_{2,\max}$ for c_2. Position $z_{g,k}$ represents the global best in the group of particles, and $\widehat{z}_{p,k}$ denotes a record of the individual particle's best solution over the past iterations, in which $p = 1, 2, \ldots, P$ is the index of the particle, and k is the iteration count.



(a) Velocity versus truck orientation and distance

(b) Velocity versus trailer orientation and distance

(c) Velocity versus truck orientation and trailer orientation

(d) Turn angle versus truck orientation and distance

(e) Turn angle versus trailer orientation and distance

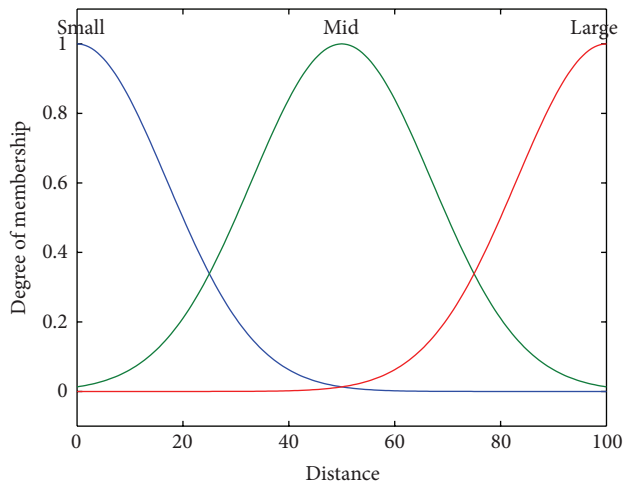
(f) Turn angle versus truck orientation and trailer orientation

Figure 6: Fuzzy rule surfaces based on human expert knowledge.

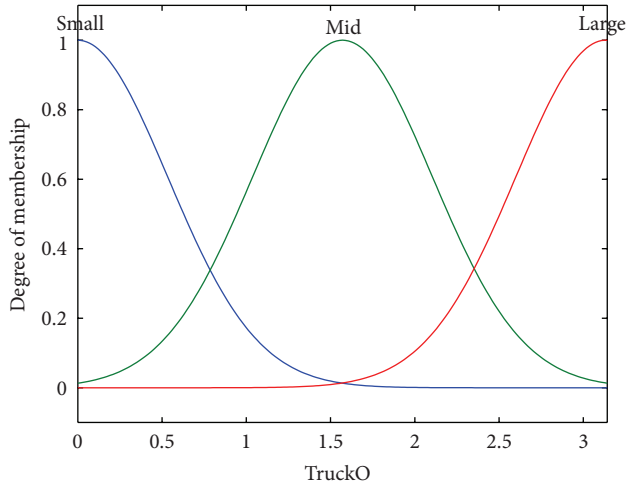
3.3. PSO for Fuzzy Logic Controller Tuning. In the Mamdani type of controller, the conventional fuzzy rules can be represented as

$$R^i = \text{IF } \left(X_1^i \text{ is } A_1^i\right) \cdots \text{AND } \left(X_j^i \text{ is } A_m^i\right) \cdots \text{AND } \left(X_{N_i}^i \text{ is } A_{N_m}^i\right) \text{ THEN } \left(Y_1^i \text{ is } B_1^i\right) \cdots \text{AND } \left(Y_k^i \text{ is } B_n^i\right) \cdots \text{AND } \left(Y_{N_o}^i \text{ is } B_{N_n}^i\right), \tag{19}$$

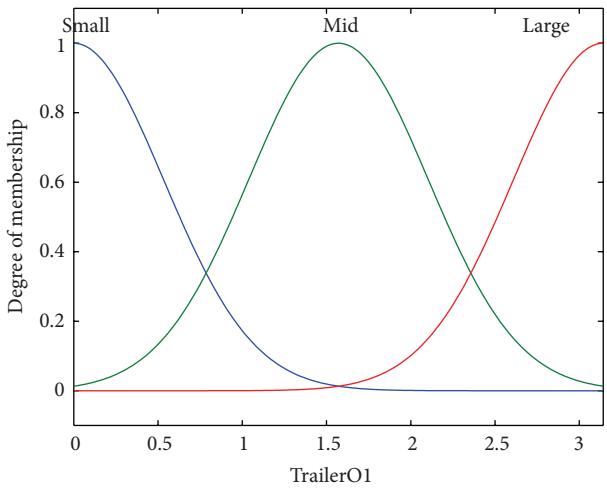
where terms X_j^i $(j = 1, 2, \ldots, N_i)$ represent the input variables and Y_k^i $(k = 1, 2, \ldots, N_0)$ represent the output variables. The numbers N_i and N_0 together represent the number of



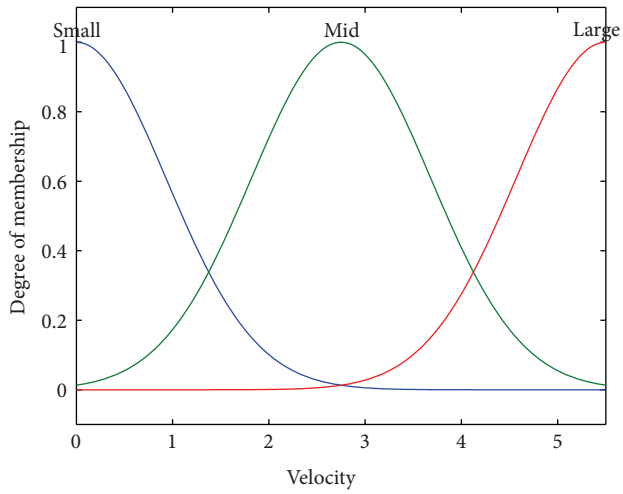
(a) Fuzzy membership functions of system distance



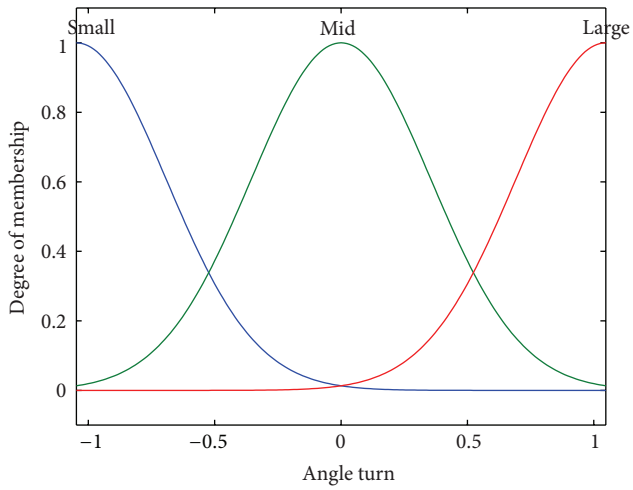
(b) Fuzzy membership functions of truck orientation



(c) Fuzzy membership functions of trailer orientation

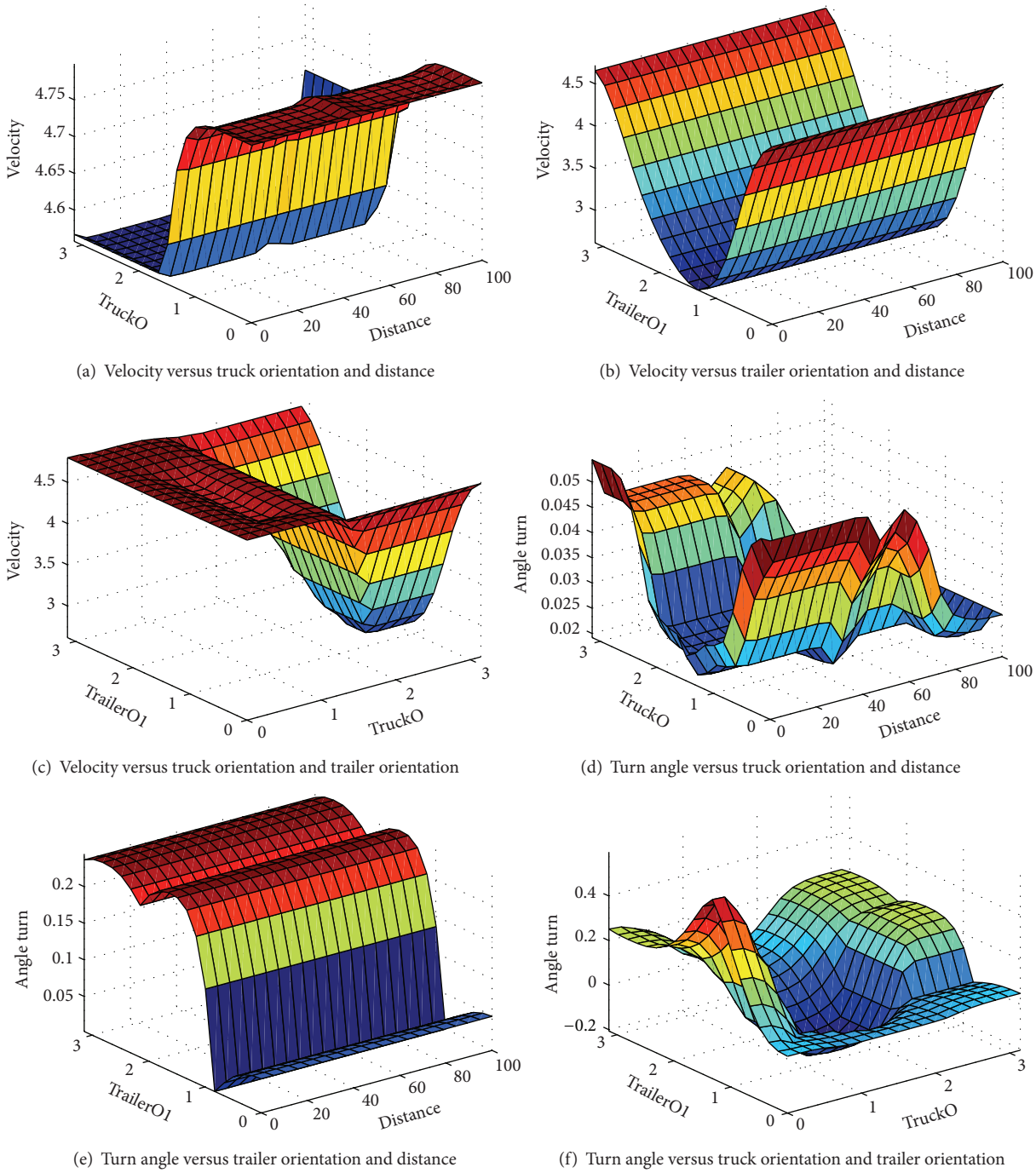


(d) Fuzzy membership functions of system velocity



(e) Fuzzy membership functions of truck turn rate

Figure 7: Fuzzy membership functions.



(a) Velocity versus truck orientation and distance

(b) Velocity versus trailer orientation and distance

(c) Velocity versus truck orientation and trailer orientation

(d) Turn angle versus truck orientation and distance

(e) Turn angle versus trailer orientation and distance

(f) Turn angle versus truck orientation and trailer orientation

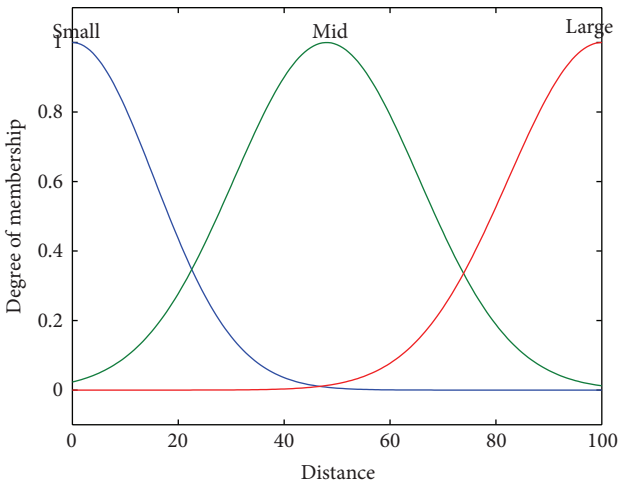
Figure 8: Fuzzy rule surfaces tuned by PSO algorithm.

input and output variables. A_m^i, B_n^i denote the corresponding linguistic values (fuzzy sets), respectively. Furthermore, $m = 1, 2, \ldots, N_m$ and $n = 1, 2, \ldots, N_n$. R^i, $i = 1, 2, \ldots, N_R$, is the ith rule; N_R is the number of rules.

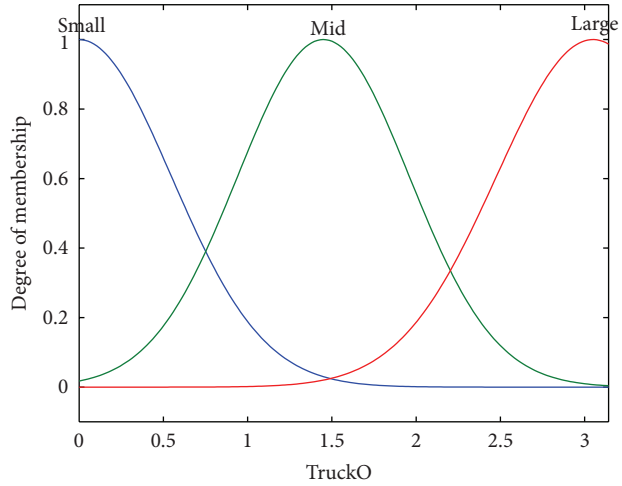
3.3.1. Fuzzy Rules Tuning. The relationship of input and output can be expressed using numerical values. Therefore, fuzzy rules could be tuned automatically by employing the PSO algorithm. Let the fuzzy rule be expressed by numerical values as follows:

$$123, 12\,(1) : 1. \tag{20}$$

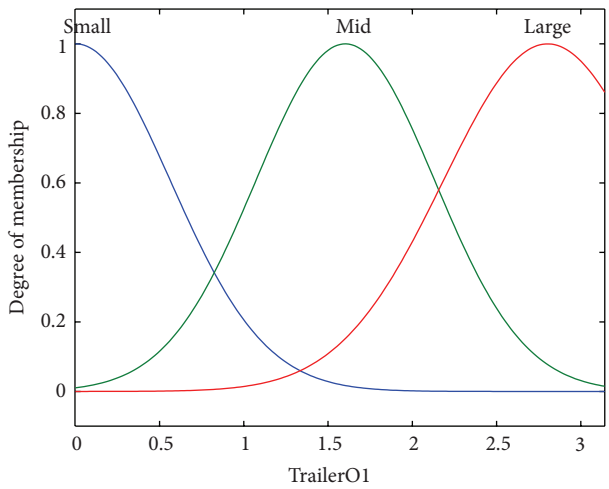
In (20), the first three digits represent the membership function associated with three inputs (1 denotes large, 2 denotes middle, and 3 denotes small). The two digits after the comma indicate two outputs (1 denotes large; 2 denotes



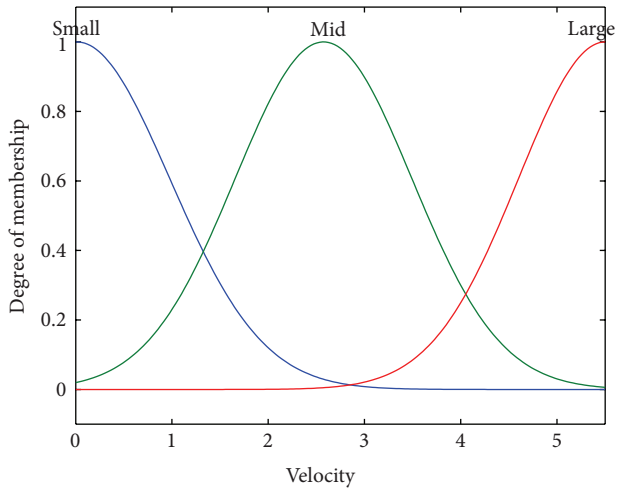
(a) PSO tuned fuzzy membership functions of system distance



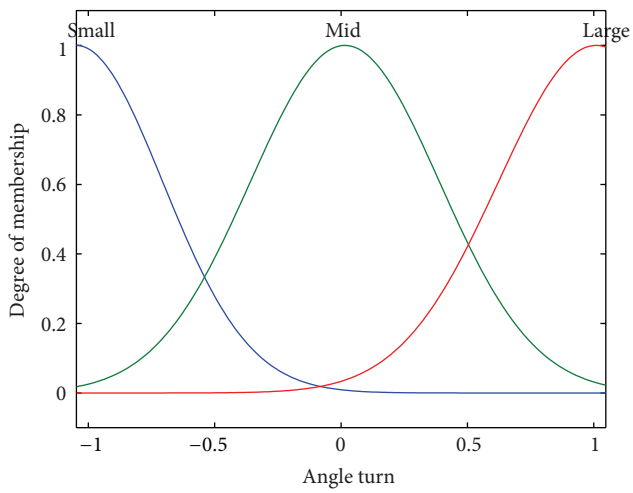
(b) PSO tuned fuzzy membership functions of truck orientation



(c) PSO tuned fuzzy membership functions of trailer orientation



(d) PSO tuned fuzzy membership functions of system velocity



(e) PSO tuned fuzzy membership functions of truck turn rate

Figure 9: PSO tuned fuzzy membership functions.

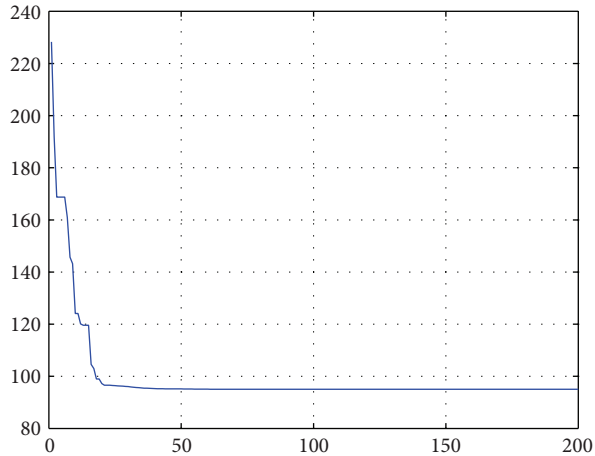


Figure 10: Control error of the truck and trailer system.

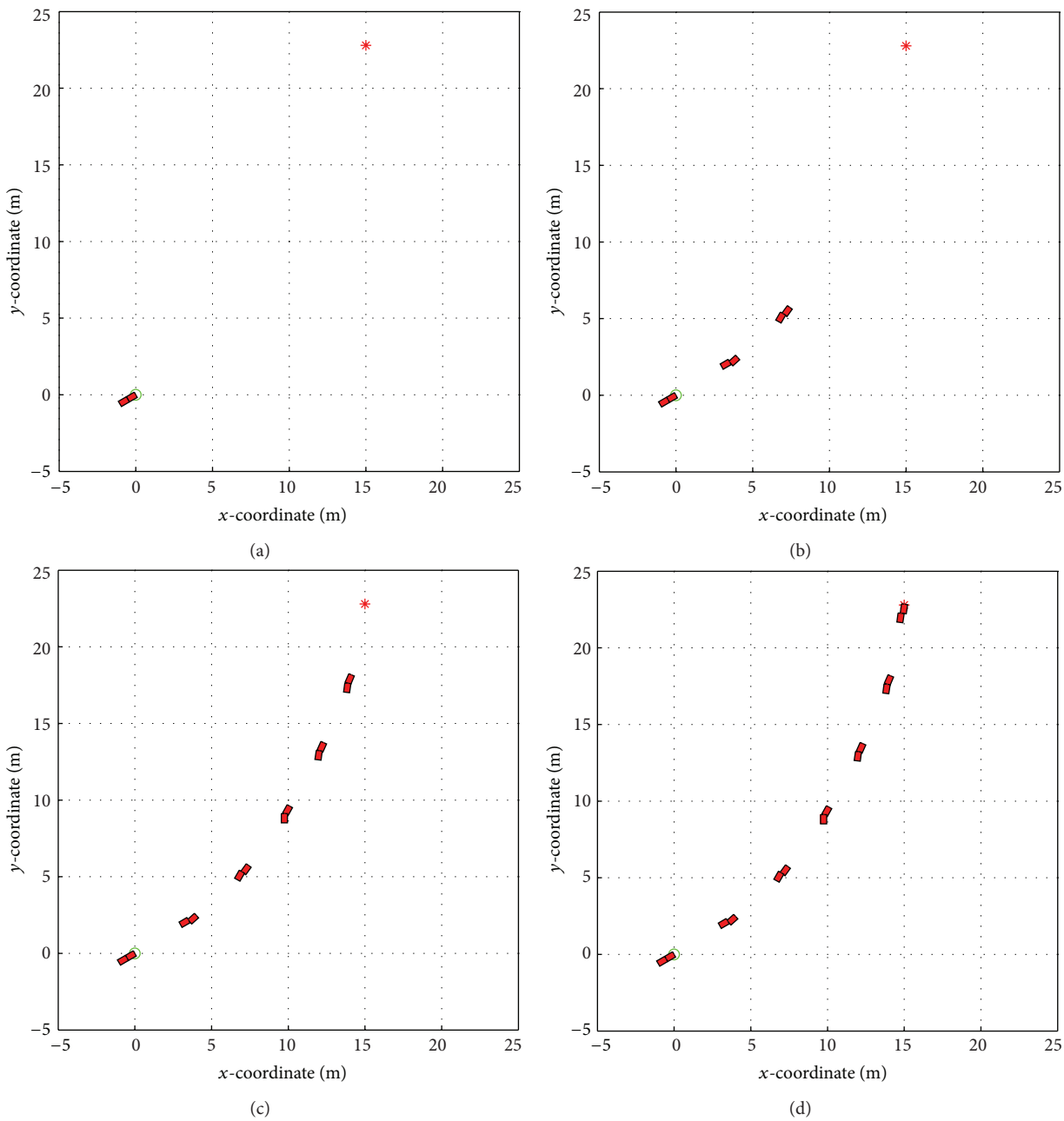


Figure 11: Motion of truck and multiple trailer system.

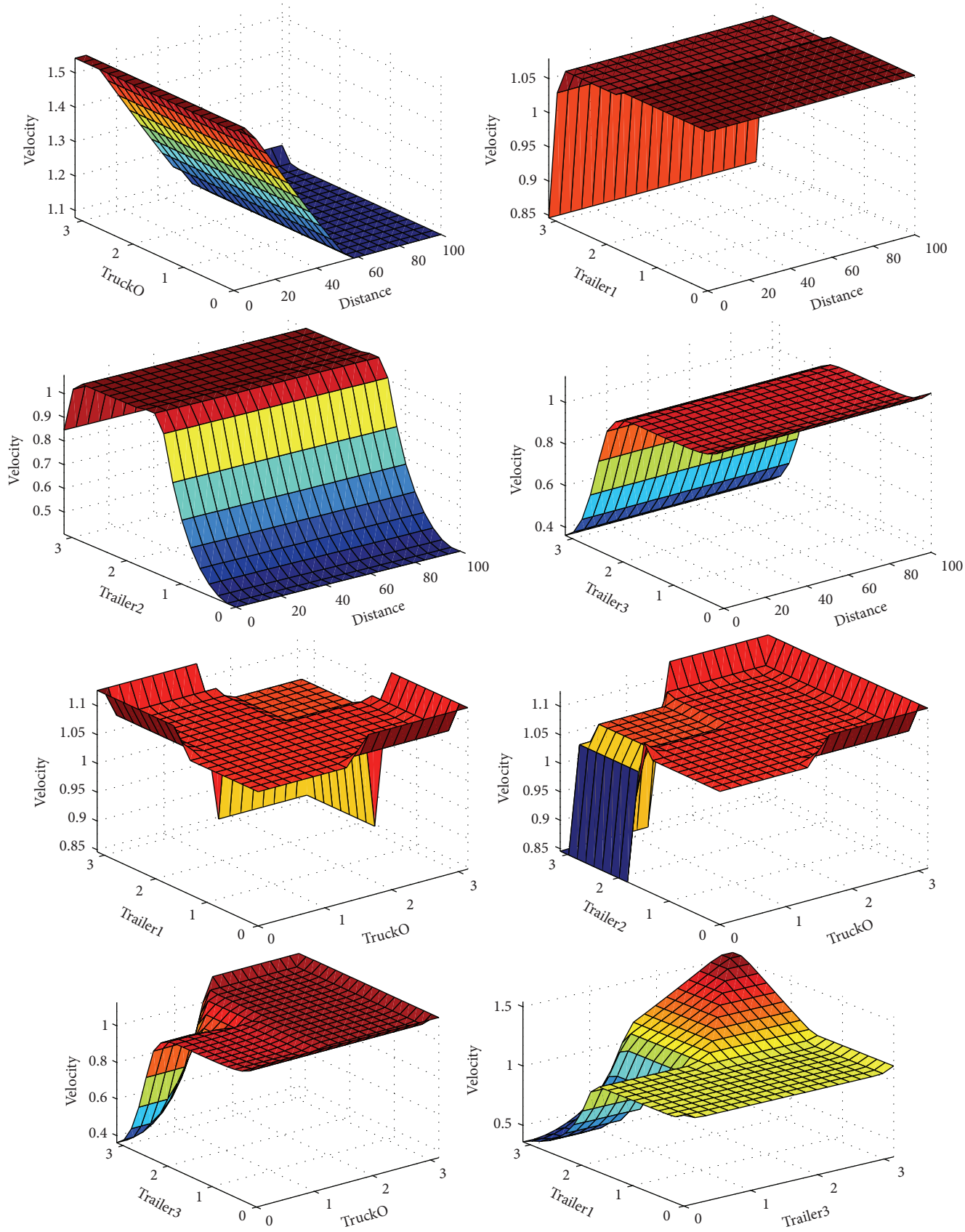


Figure 12: Continued.

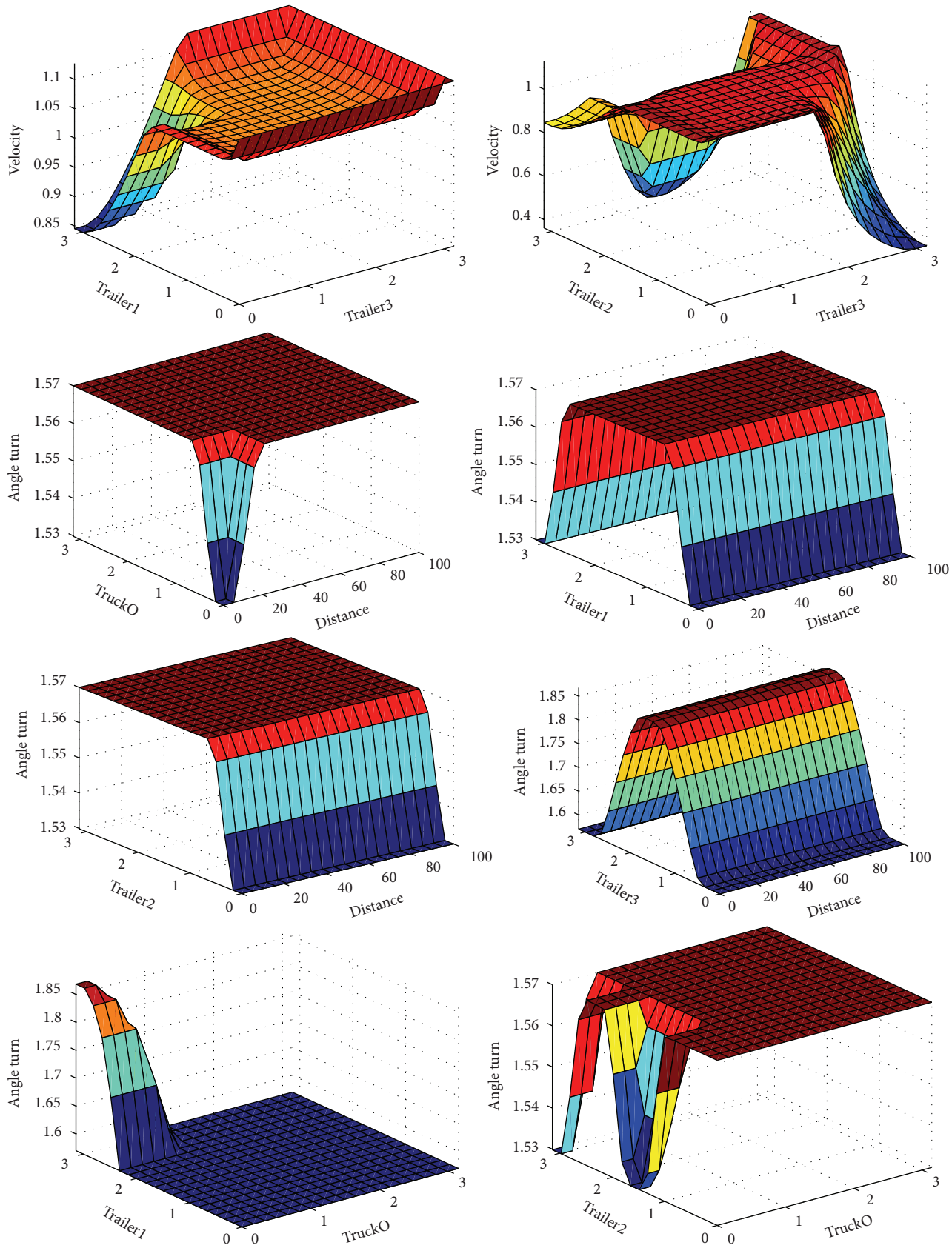


Figure 12: Continued.

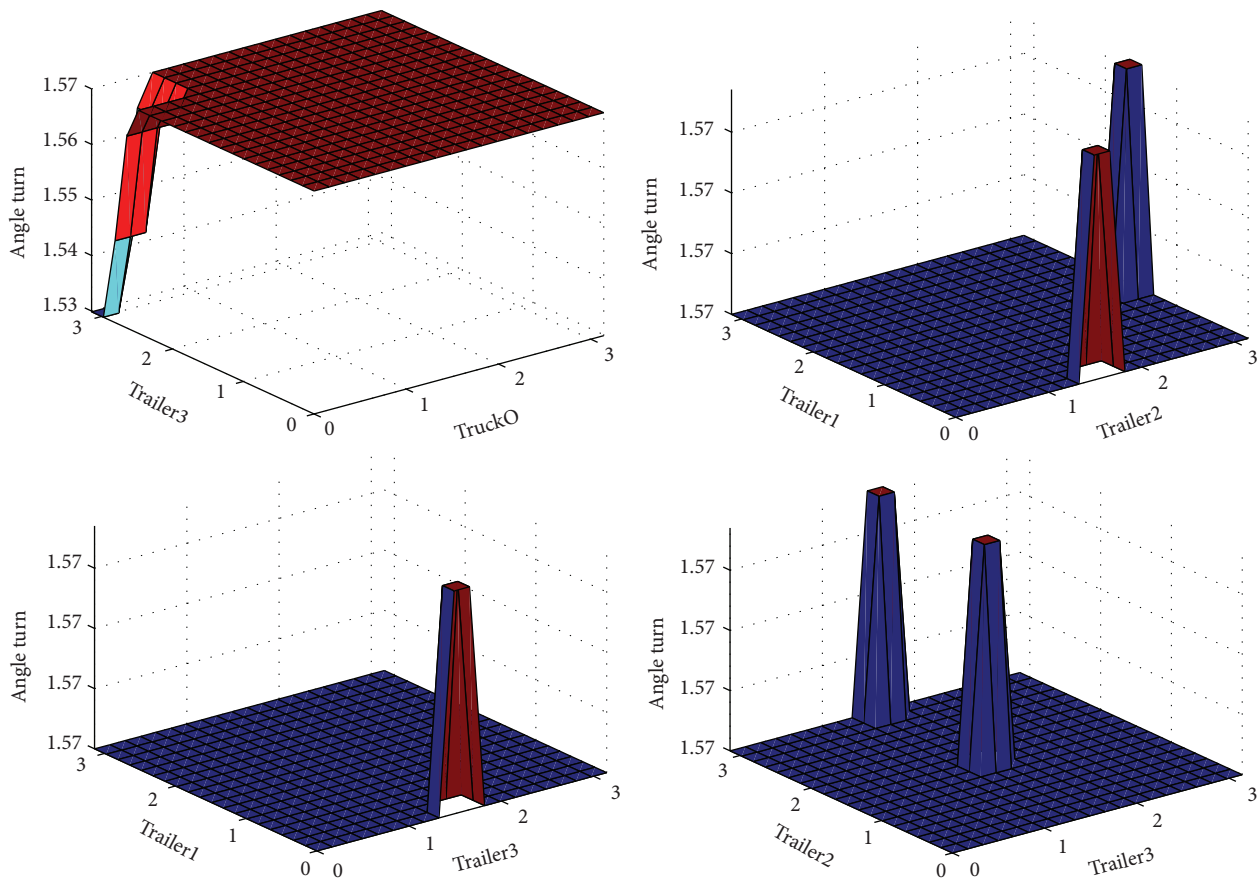


Figure 12: Randomly chosen fuzzy rule surfaces of truck and multiple trailer system.

middle). The value in the bracket is the weighting factor of this rule while the last digit after the colon represents the relationship between the three inputs (1 denotes AND; 2 denotes OR). Therefore, it is possible to find the appropriate relationship between the rules and to determine if the rules are adequate.

3.3.2. Fuzzy Membership Functions Tuning. The membership functions employed in this fuzzy logic controller are all of Gaussian forms. The Gaussian (the normal distribution) is a continuous probability distribution that has a bell-shaped appearance. It is generally considered as the most prominent probability distribution in statistics. The parameters that define the membership functions are the mean c and deviation σ of each membership function, which is defined as:

$$f(x, \sigma, c) = \exp\left\{-\frac{(x-c)^2}{2\sigma^2}\right\}. \tag{21}$$

Figure 5 shows the strategy of using a PSO for rule tuning in fuzzy logic controller. In the proposed PSO process, each particle is formed to represent the rule of the fuzzy logical controller. The purpose of the PSO is to find the set of rules which minimize the response error of the truck and trailer system. The objective function is defined as

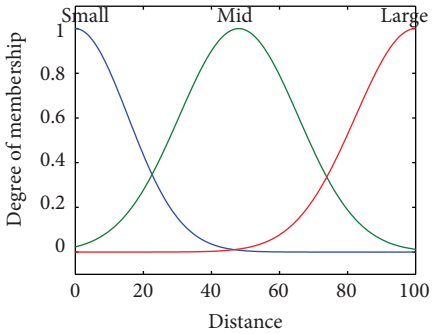
$$\text{fit} = \sum_{t=0}^{T} E_{\text{total}}^2, \tag{22}$$

where T is the total running time of the fuzzy logic controller, E_{total} is the error between the reference inputs and actual system outputs.

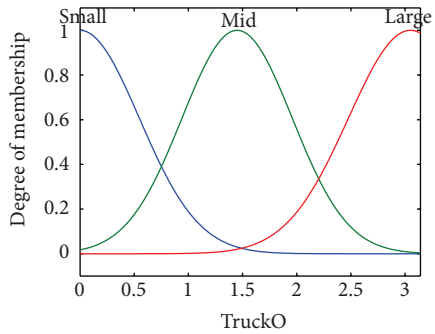
4. Simulation

In this section, a PSO tuned fuzzy logic controller is designed for steering the truck-and-multi-trailer system to its desired position. The objective function that evaluates the fitness of each practice was given in (22). After tuning membership functions and fuzzy rules, the fuzzy logic controller is able to provide a minimized response error. In order to verify the effectiveness of the proposed method, computer simulation results of the TMT system control are shown in this section. Simulations are conducted with both one-trailer system and multiple trailer system.

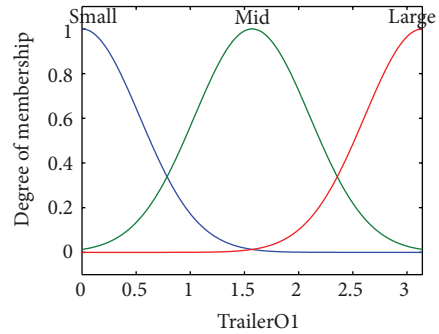
In practical applications the truck could start from arbitrary locations and orientations and is steered by the drive commands to the desired position. In the simulation studies, it is assumed that the truck starts from the origin of the world



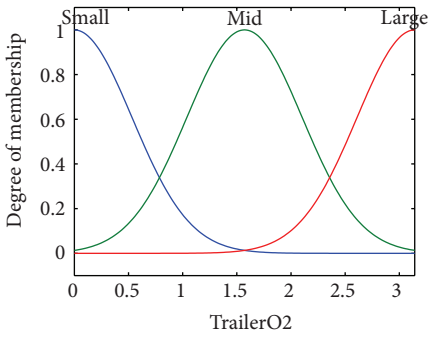
(a) Fuzzy membership functions of system distance



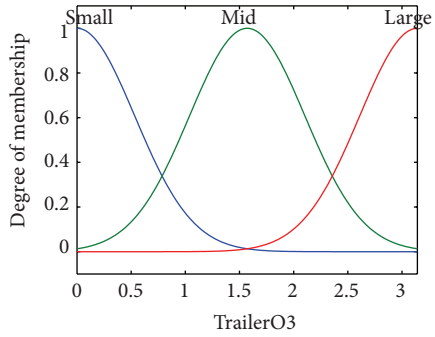
(b) Fuzzy membership functions of truck orientation



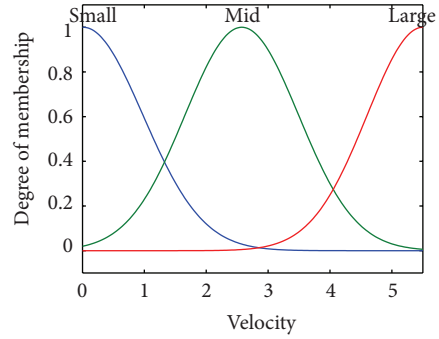
(c) Fuzzy membership functions of trailer one orientation



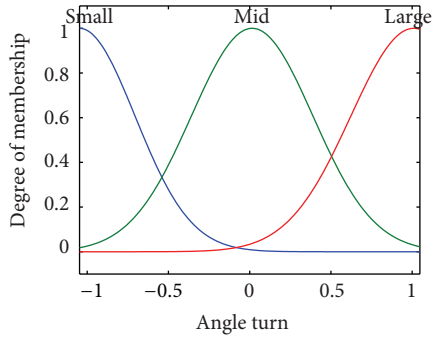
(d) Fuzzy membership functions of trailer two orientation



(e) Fuzzy membership functions of trailer three orientation



(f) Fuzzy membership functions of system velocity



(g) Fuzzy membership functions of truck turn rate

Figure 13: Fuzzy membership functions of truck and multiple trailer system.

coordinate and parallel with x-axis; that is, $(x_0, y_0) = 0$. The truck and trailer are initialized with an orientation of $\pi/6$ to the world coordinate. The desired position is selected randomly, and the desired orientations for both the truck and the trailer is $\pi/2$. Furthermore, the maximum linear velocity of the truck is $v_{\max} = 0.1$ m/s. The lengths of the vehicles are $F_i + R_i = 0.2$ m.

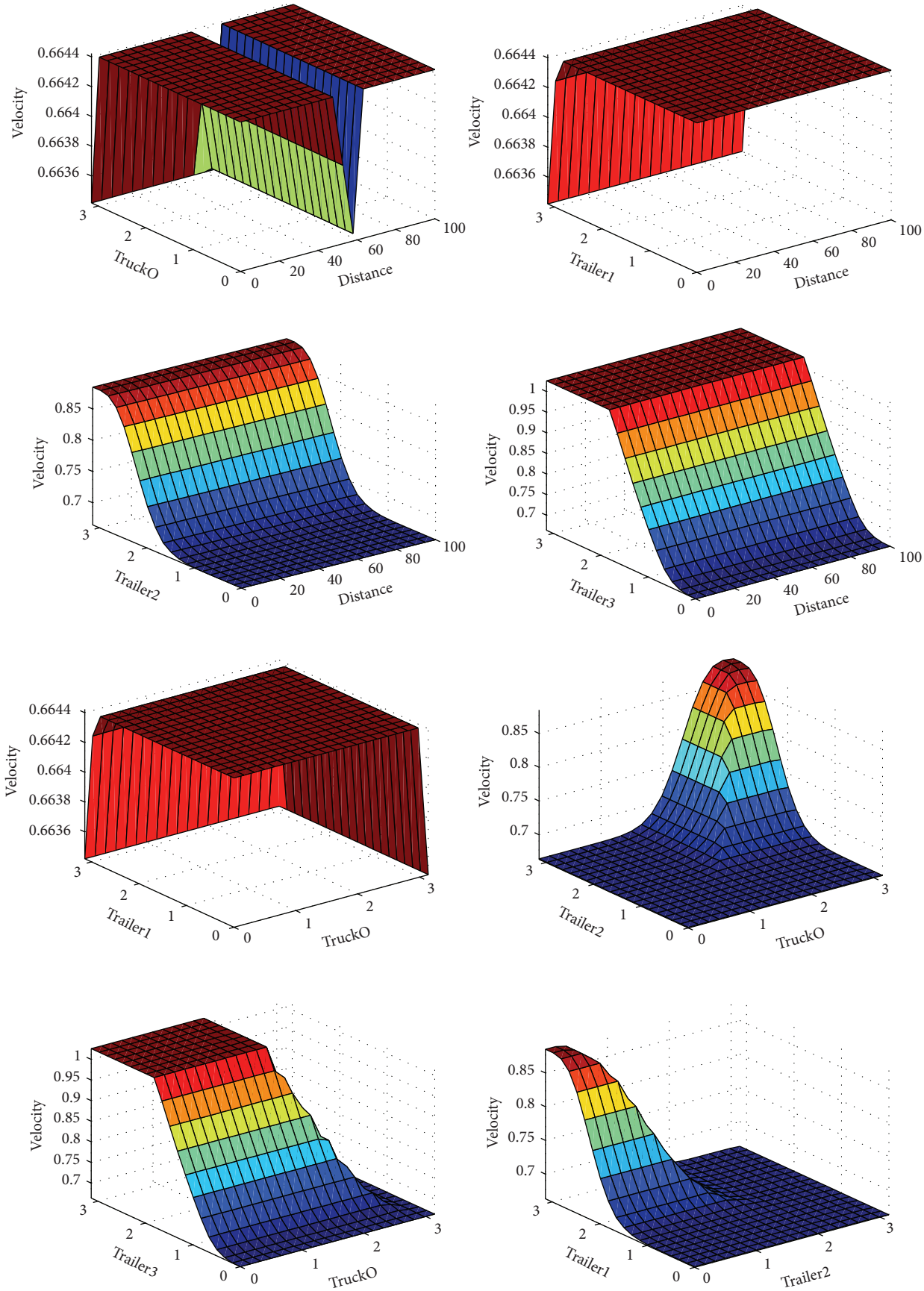
4.1. Case Study: One-Trailer System. For the one-trailer system, there are three inputs to the controller. The first input, labeled Distance, is the distance between the reference input and the current truck position. The second input, labeled TruckO (truck orientation) is the orientation of the truck with respect to x-axis. The third input represents the orientation of trailer (TruckO1) with respect to x-axis. The output of the fuzzy logical controller is the linear velocity and turn rate while moving the system.

The initial controller is set to have 10 rules:

R^1: IF (Distance is Small) AND (TruckO is Small) AND (TrailerO1 is Large) THEN (Velocity is Small) AND (Turning rate is Mid).

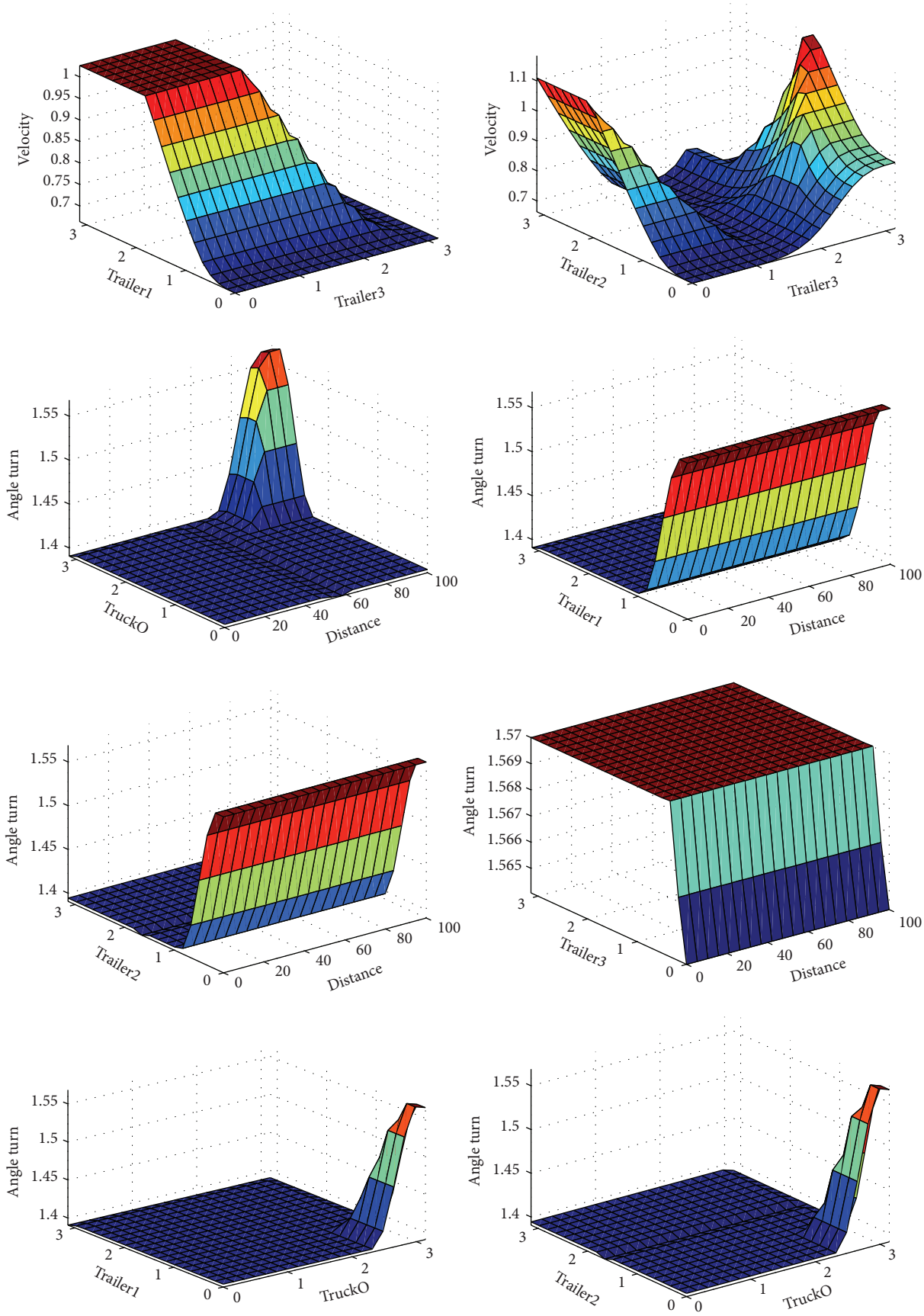
R^2: IF (Distance is Mid) AND (TruckO is Small) AND (TrailerO1 is Large) THEN (Velocity is Small) AND (Turning rate is Mid).

R^3: IF (Distance is Small) AND (TruckO is Large) AND (TrailerO1 is Mid) THEN (Velocity is Mid) AND (Turning rate is Large).



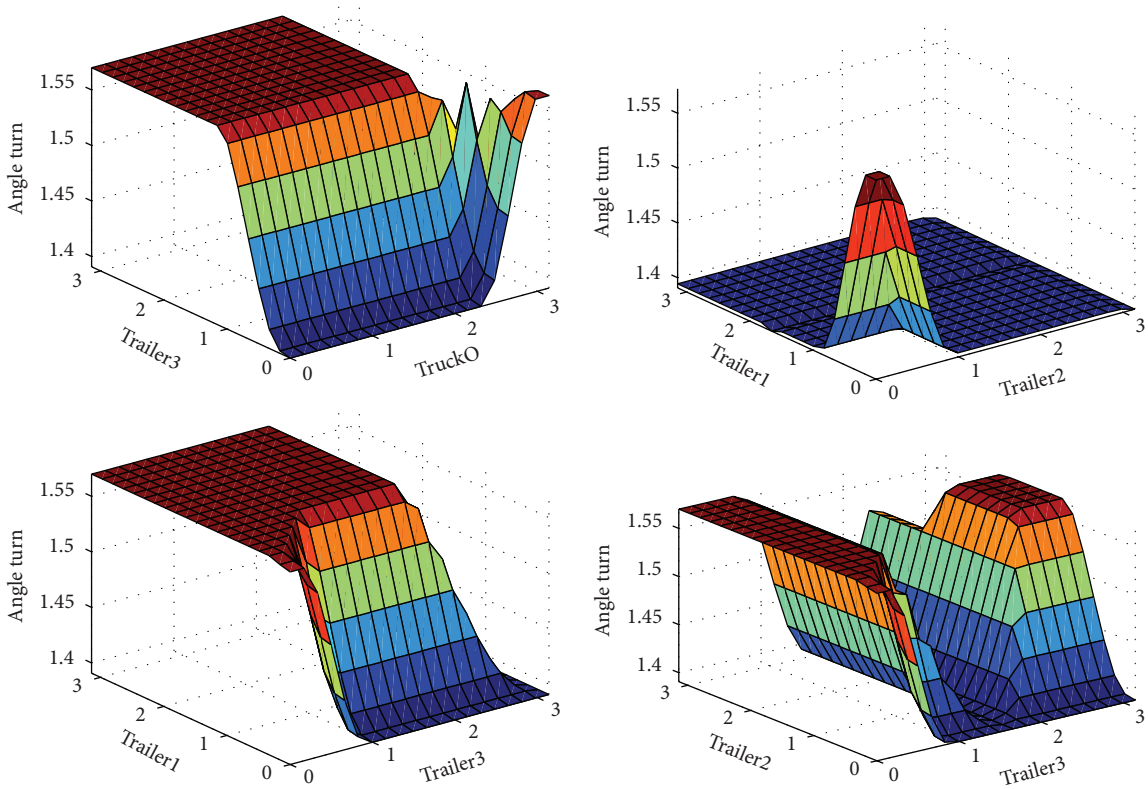
(a)

Figure 14: Continued.



(b)

Figure 14: Continued.



(c)

Figure 14: PSO tuned fuzzy membership functions.

R^4: IF (Distance is Large) and (TruckO is Mid) AND (TrailerO1 is Mid) THEN (Velocity is Large) AND (Turning rate is Small).

R^5: IF (Distance is Large) AND (TruckO is small) AND (TrailerO1 is Large) THEN (Velocity is Large) AND (Turning rate is Small).

R^6: IF (Distance is Large) AND (TruckO is Mid) AND (TrailerO1 is Large) THEN (Velocity is Large) AND (Turning rate is Mid).

R^7: IF (Distance is Large) AND (TruckO is Large) AND (TrailerO1 is Mid) THEN (Velocity is Large) AND (Turning rate is Large).

R^8: IF (Distance is Large) AND (TruckO is Small) AND (TrailerO1 is Small) THEN (Velocity is Large) AND (Turning rate is Small).

R^9: IF (Distance is Mid) AND (TruckO is Mid) AND (TrailerO1 is Mid) THEN (Velocity is Mid) AND (Turning rate is Mid).

R^{10}: IF (Distance is Mid) AND (TruckO is Small) AND (TrailerO1 is Small) THEN (Velocity is Mid) AND (Turning rate is Small).

The fuzzy rule surfaces are showed in Figure 6. Initial fuzzy membership functions are showed in Figure 7.

In the particle swarm optimization algorithm, the total searching iterations are set to be 200. It is also defined in the searching process that if the error is not reduced for 10 iterations then the search should be terminated. The inertia factor w was set to be 0.5, and weighting factors c_1 and c_2 were set to be 1 and 0.2, respectively. At the completion of the PSO tuning, the rules of the fuzzy controller have attended a reduced number. The final rules are given by the following:

R^1: IF (Distance is Large) OR (TruckO is Large) OR (TrailerO1 is Large) THEN (Velocity is Small) AND (Turning rate is Small).

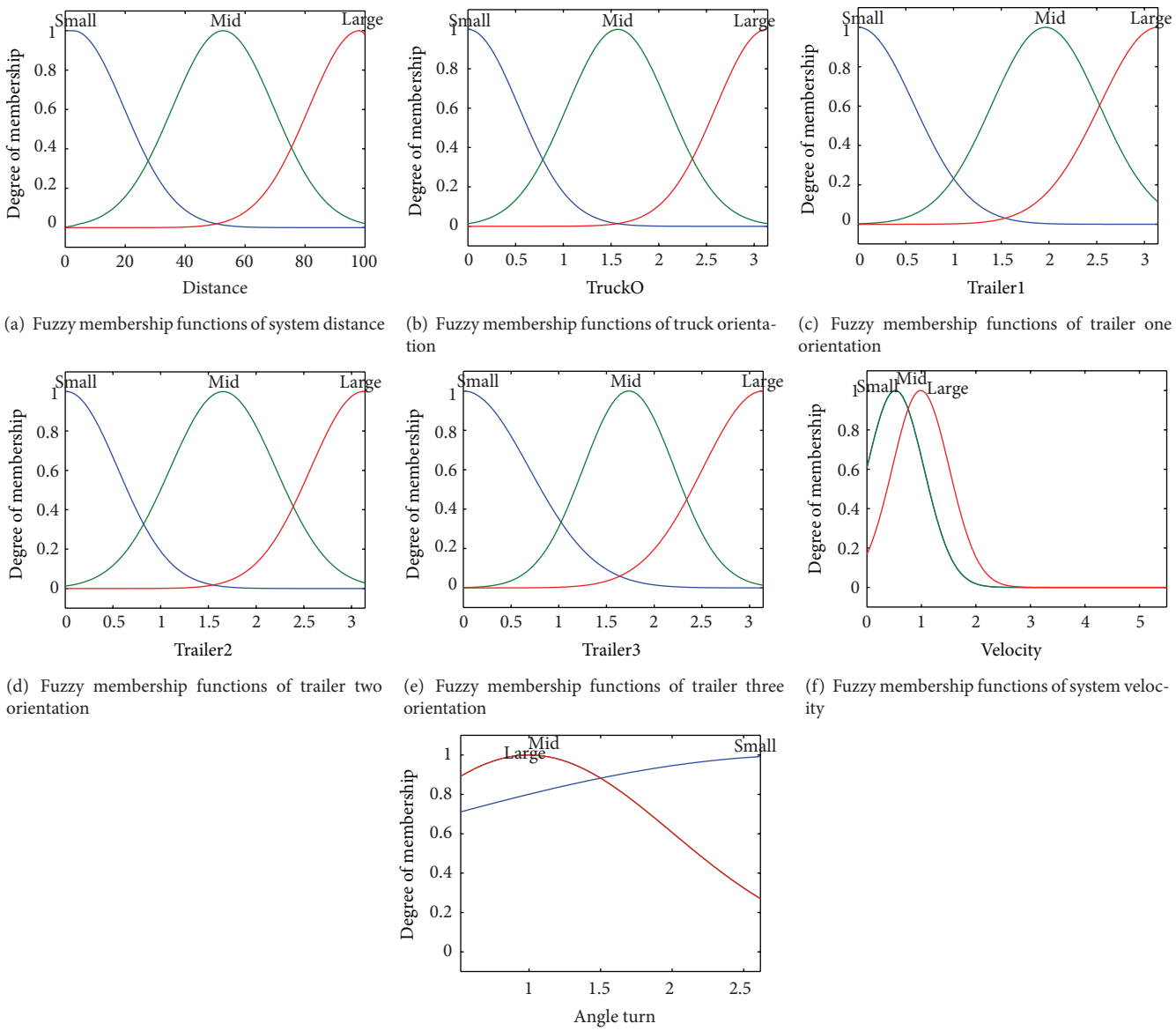
R^2: IF (Distance is Large) OR (TruckO is Mid) OR (TrailerO1 is Mid) THEN (Velocity is Large) AND (Turning rate is Small).

R^3: IF (Distance is Small) OR (TruckO is Large) OR (TrailerO1 is Mid) THEN (Velocity is Large) AND (Turning rate is large).

R^4: IF (Distance is Large) AND (TruckO is Small) AND (TrailerO1 is Small) THEN (Velocity is Large) AND (Angle Turn is Large).

R^5: IF (Distance is Small) AND (TruckO is Small) AND (TrailerO1 is Large) THEN (Velocity is Large) AND (Angle Turn is Small).

R^6: IF (Distance is Small) OR (TruckO is Mid) OR (TrailerO1 is Large) THEN (Velocity is Large) AND (Angle Turn is Large).



(a) Fuzzy membership functions of system distance

(b) Fuzzy membership functions of truck orientation

(c) Fuzzy membership functions of trailer one orientation

(d) Fuzzy membership functions of trailer two orientation

(e) Fuzzy membership functions of trailer three orientation

(f) Fuzzy membership functions of system velocity

(g) Fuzzy membership functions of truck tune rate

Figure 15: PSO tuned fuzzy membership functions of TMT system.

Generally, the number of fuzzy rules that have to be processed for fuzzy controller to make decision mainly depends on the number of input variables, the number of output variables, and the number of predefined linguistic values for an individual input variable. Therefore, the total number of the fuzzy rules can be calculated by the following equation:

$$N_R = N_m^{N_i} \times N_n^{N_o}, \tag{23}$$

where N_i and N_o denote the number of inputs and outputs respectively, and N_m and N_n are the linguistic values of the input and output. Based on (23), the total rule number of the fuzzy controller is 243. However, after PSO tuning, only 6 rules are employed in this fuzzy controller if weighting factor approaches zero; this rule is deleted. Figure 8 shows the rule surfaces which are generated by the PSO rule tuned fuzzy logic controller.

At the completion of the PSO membership functions tuning, the membership functions for the inputs and outputs of the fuzzy logic controller have been modified. They are shown in Figure 9.

Based on the rules which are generated by the PSO algorithm, the control error is shown in Figure 10. The maximum error has been reduced from 229.45 to be around 96.02 after 200 iterations.

Figure 11 is the simulation result of the truck and multiple trailer system. The truck is initially located at the origin of the coordinate system as shown in Figure 11(a) and then moves to

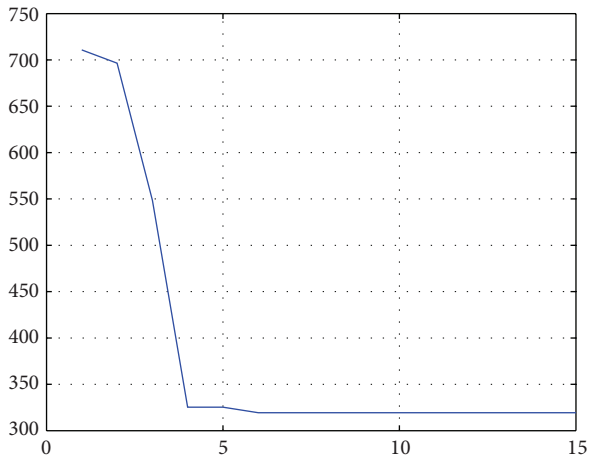


Figure 16: Control error of the truck and trailer system.

the desired position which is shown in Figures 11(b) and 11(c). In Figure 11(d), the system arrived at the end point with the minimum control error.

4.2. Case Study: Multiple Trailer System. For a system with three trailers, there are five inputs to the controller. The first input, labeled Distance, is the distance between the reference input and the current truck position. The second input, labeled TruckO (truck orientation) is the orientation of the truck with respect to x-axis. The rest of inputs present the orientation of three trailers (TrailerO1, TrailerO2, and TrailerO3) with respect to x-axis. The output of the fuzzy logical controller is the linear velocity and turn rate to drive the system.

For multiple trailer system, it is extremely difficult to build fuzzy rules based on human expert knowledge. Therefore, the initial 10 fuzzy rules are randomly set as follows:

R^1: IF (Distance is Large) AND (TruckO is Large) AND (TrailerO1 is Large) AND (TrailerO2 is Small) AND (TrailerO3 is Small) THEN (Velocity is Small) AND (Turning rate is Small).

R^2: IF (Distance is Large) AND (TruckO is Large) AND (TrailerO1 is Large) AND (TrailerO2 is Small) AND (TrailerO3 is Large) THEN (Velocity is Large) AND (Turning rate is Large).

R^3: IF (Distance is Small) AND (TruckO is Mid) AND (TrailerO1 is Mid) AND (TrailerO2 is Small) AND (TrailerO3 is Small) THEN (Velocity is Small) AND (Turning rate is Small).

R^4: IF (Distance is Small) AND (TruckO is Mid) AND (TrailerO1 is Mid) AND (TrailerO2 is Large) AND (TrailerO3 is Small) THEN (Velocity is Large) AND (Turning rate is Small).

R^5: IF (Distance is Small) AND (TruckO is Small) AND (TrailerO1 is Small) AND (TrailerO2 is Mid) AND (TrailerO3 is Large) THEN (Velocity is Small) AND (Turning rate is Large).

R^6: IF (Distance is Small) AND (TruckO is Large) AND (TrailerO1 is Large) AND (TrailerO2 is Small) AND (TrailerO3 is Large) THEN (Velocity is Small) AND (Turning rate is Large).

R^7: IF (Distance is Small) AND (TruckO is Small) AND (TrailerO1 is Small) AND (TrailerO2 is Small) AND (TrailerO3 is Small) THEN (Velocity is Mid) AND (Turning rate is Mid).

R^8: IF (Distance is Mid) AND (TruckO is Large) AND (TrailerO1 is Large) AND (TrailerO2 is Large) AND (TrailerO3 is Mid) then (Velocity is Small) AND(Turning rate is Large).

R^9: IF (Distance is Small) AND (TruckO is Small) AND (TrailerO1 is Small) AND (TrailerO2 is Small) AND (TrailerO3 is Mid) then (Velocity is Small) AND (Turning rate is Mid).

R^{10}: IF (Distance is Mid) AND (TruckO is Small) AND (TrailerO1 is Small) AND (TrailerO2 is Large) AND (TrailerO3 is Small) THEN (Velocity is Mid) AND (Turning rate is Small).

Rule surfaces are shown in Figure 12 and Initial fuzzy membership functions are shown in Figure 13.

At the completion of the PSO membership functions tuning, the fuzzy rule surface are shown in Figure 14. The membership functions for the inputs and outputs of the fuzzy logic controller have been modified. They are shown in Figure 15.

Based on the rules which are generated by the PSO algorithm, the control error is shown in Figure 16. The maximum error has been reduced from 729.18 to be around 330.14 after 15 iterations.

Simulation results of the truck-and-multi-trailer system are shown below. Figure 17(a) indicates the TMT system which is initially located at the origin of the coordinate system. In Figures 17(b) and 17(c), the system is driven to the desired position. Figure 17(d) shows the system arrived to the end point with the minimum control error.

5. Conclusion

In fuzzy logic controller design, there are large numbers of fuzzy rules that have to be processed in order to produce decisions. The number of rules in a fuzzy controller primarily originates from the number of input variables that are entering the decision process, and one possible solution for decreasing it is to use the method of decomposition. In addition, the process of tuning fuzzy membership functions is also time consumed and often frustrated. In this paper a particle swarm optimization based automatic fuzzy rules and fuzzy membership function tuning method in designing fuzzy logic controller is introduced to control a truck-and-multi-trailer system. Simulation results for both one-trailer system and multiple trailer system have shown that the fuzzy

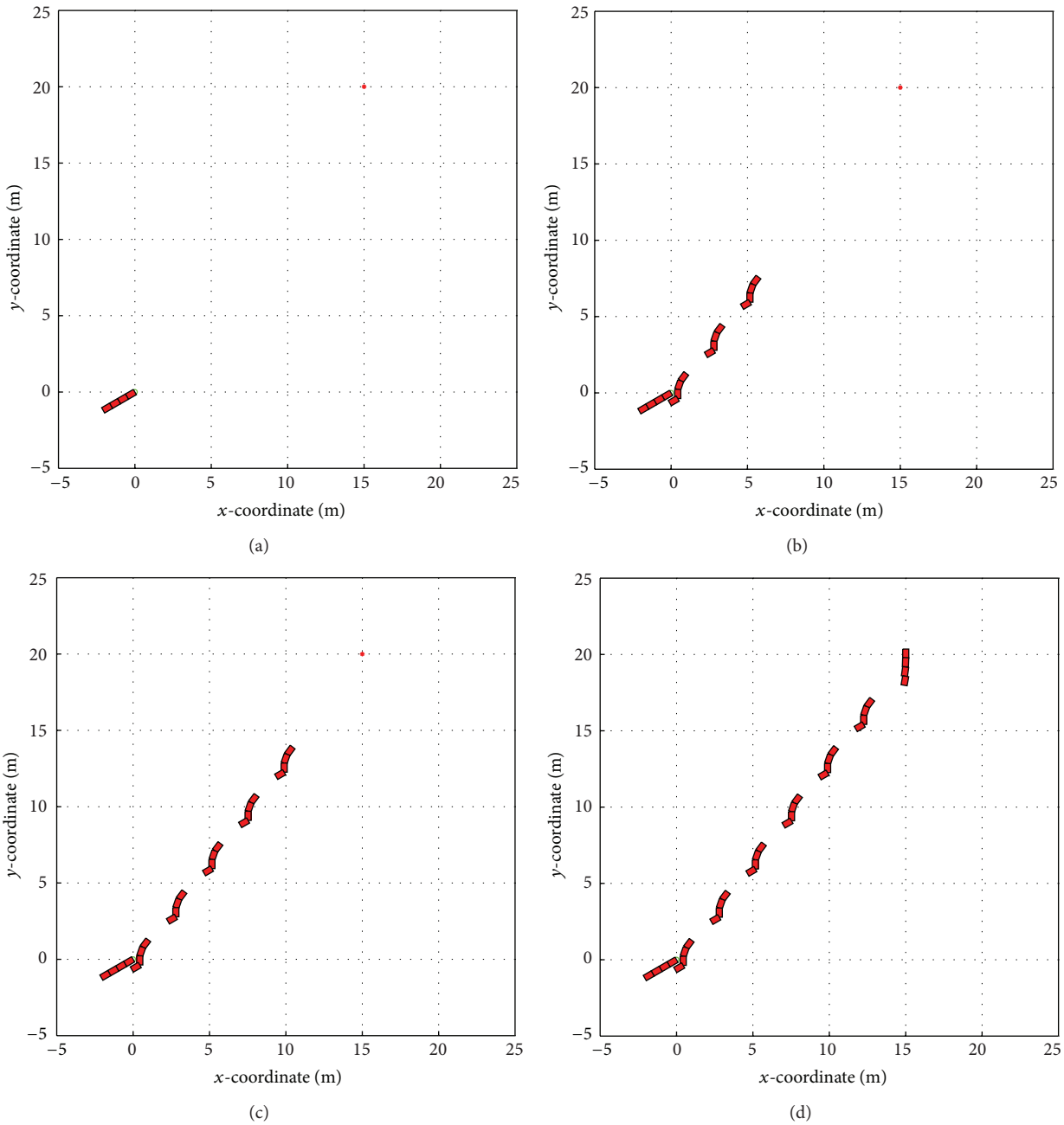


Figure 17: Motion of truck and multiple trailer system.

rules have been decreased, and tuned membership function is performing satisfactorily result.

References

[1] F. Lamiraux, J.-P. Laumond, C. van Geem, D. Boutonnet, and G. Raust, "Trailer-truck trajectory optimization: the transportation of components for the airbus A380," *IEEE Robotics and Automation Magazine*, vol. 12, no. 1, pp. 14–21, 2005.

[2] J. I. Roh, H. Lee, and W. Chung, "Control of a car with a trailer using the driver assistance system," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO '11)*, pp. 2890–2895, 2011.

[3] W. Li, T. Tsubouchi, and S. Yuta, "Manipulative difficulty of a mobile robot pushing and towing multiple trailers," *Advanced Robotics*, vol. 14, no. 3, pp. 169–183, 2000.

[4] J.-H. Lee, W. Chung, M. Kim, and J.-B. Song, "A passive multiple trailer system with off-axle hitching," *International Journal of Control, Automation and Systems*, vol. 2, no. 3, pp. 289–297, 2004.

[5] C. Altafini, "Some properties of the general n-trailer," *International Journal of Control*, vol. 74, no. 4, pp. 409–424, 2001.

[6] J. L. Martinez, J. Morales, A. Mandow, and A. Garcia-Cerezo, "Steering limitations for a vehicle pulling passive trailers," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 4, pp. 809–818, 2008.

[7] K. Tanaka, K. Yamauchi, H. Ohtake, and H. O. Wang, "Sensor reduction for backing-up control of a vehicle with triple trailers," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 2, pp. 497–509, 2009.

[8] K. Matsushita and T. Murakami, "Backward motion control for articulated vehicles with double trailers considering driver's input," in *Proceedings of the 32nd Annual Conference on IEEE Industrial Electronics (IECON '06)*, pp. 3052–3057, November 2006.

[9] J. Cheng, Y. Zhang, S. Hou, and B. Song, "Stabilization control of a backward tractor-trailer mobile robot," in *Proceedings of the 8th World Congress on Intelligent Control and Automation (WCICA '10)*, pp. 2136–2141, July 2010.

[10] M. Park, W. Chung, M. Kim, and J. Song, "Control of a mobile robot with passive multiple trailers," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 5, pp. 4369–4374, May 2004.

[11] K. Yoo and W. Chung, "Pushing motion control of n passive off-hooked trailers by a car-like mobile robot," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '10)*, pp. 4928–4933, May 2010.

[12] W. Chung, M. Park, K. Yoo, J. I. Roh, and J. Choi, "Backward-motion control of a mobile robot with n passive off-hooked trailers," *Journal of Mechanical Science and Technology*, vol. 25, no. 11, pp. 2895–2905, 2011.

[13] J. Yuan and Y. Huang, "Path following control for tractor-trailer mobile robots with two kinds of connection structures," in *Proceddings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '06)*, pp. 2533–2538, October 2006.

[14] C. Pradalier and K. Usher, "Robust trajectory tracking for a reversing tractor trailer," *Journal of Field Robotics*, vol. 25, no. 6-7, pp. 378–399, 2008.

[15] Z. Leng and M. Minor, "A simple tractor-trailer backing control law for path following," in *Proceedings of the 23rd IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '10)*, pp. 5538–5542, October 2010.

[16] M. Michałek, "Tracking control strategy for the standard N-trailer mobile robot—a geometrically motivated approach," *Lecture Notes in Control and Information Sciences*, vol. 422, pp. 39–51, 2012.

[17] S. Wen, W. Zheng, J. Zhu, X. Li, and S. Chen, "Elman fuzzy adaptive control for obstacle avoidance of mobile robots using hybrid force/position incorporation," *IEEE Transactions on Systems, Man and Cybernetics C*, vol. 42, no. 4, pp. 603–608, 2012.

[18] S.-G. Kong and B. Kosko, "Adaptive fuzzy systems for backing up a truck-and-trailer," *IEEE Transactions on Neural Networks*, vol. 3, no. 2, pp. 211–223, 1992.

[19] K. Tanaka, T. Taniguchi, and H. O. Wang, "Fuzzy controller and observer design for backing control of a trailer-truck," *Engineering Applications of Artificial Intelligence*, vol. 10, no. 5, pp. 441–452, 1997.

[20] K. Tanaka and T. Kosaki, "Design of a stable fuzzy controller for an articulated vehicle," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 27, no. 3, pp. 552–558, 1997.

[21] J. Cheng, Y. Zhang, and Z. Wang, "Curve path tracking control for tractor-trailer mobile robot," in *Proceedings of the 8th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '11)*, vol. 1, pp. 502–506, July 2011.

[22] R. El Harabi, S. B. Ali Naoui, and M. N. Abdelkrim, "Fuzzy control of a mobile robot with two trailers," in *Proceedings of the 1st International Conference on Renewable Energies and Vehicular Technology (REVET '12)*, pp. 256–262, 2012.

[23] L. X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, no. 6, pp. 1414–1427, 1992.

[24] J. I. Park, C. K. Song, J. H. Cho, and M. G. Chun, "Neuro-fuzzy rule generation for backing up navigation of car-like mobile robots," *International Journal of Fuzzy Systems*, vol. 11, no. 3, pp. 192–201, 2009.

[25] Y. Zheng, H. Shi, and S. Chen, "Fuzzy combinatorial optimization with multiple ranking criteria: a staged tabu search framework," *Pacific Journal of Optimization*, vol. 8, no. 3, pp. 457–472, 2012.

[26] O. Cordón, "A historical review of evolutionary learning methods for Mamdani-type fuzzy rule-based systems: designing interpretable genetic fuzzy systems," *International Journal of Approximate Reasoning*, vol. 52, no. 6, pp. 894–913, 2011.

[27] I. Dumitrache and C. Buiu, "Genetic learning of fuzzy controllers," *Mathematics and Computers in Simulation*, vol. 49, no. 1-2, pp. 13–26, 1999.

[28] P. C. Shill, M. A. Hossain, M. F. Amin, and K. Murase, "An adaptive fuzzy logic controller based on real coded quantum-inspired evolutionary algorithm," in *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ '11)*, pp. 614–621, June 2011.

[29] N. M. Kwok, D. K. Liu, K. C. Tan, and Q. P. Ha, "An empirical study on the settings of control coefficients in particle swarm optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 823–830, July 2006.

[30] N. M. Kwok, Q. P. Ha, D. K. Liu, G. Fang, and K. C. Tan, "Efficient particle swarm optimization: a termination condition based on the decision-making approach," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '07)*, pp. 3353–3360, September 2007.

[31] N. M. Kwok, Q. P. Ha, and G. Fang, "Motion coordination for construction vehicles using swarm intelligence," *International Journal of Advanced Robotic Systems*, vol. 4, no. 4, pp. 469–476, 2007.

[32] D. Wang, N. M. Kwok, D. K. Liu, and Q. P. Ha, "Ranked pareto particle swarm optimization for mobile robot motion planning," *Studies in Computational Intelligence*, vol. 177, pp. 97–118, 2009.

[33] A. F. M. Huang, S. J. H. Yang, M. Wang, and J. J. P. Tsai, "Improving fuzzy knowledge integration with particle swarmoptimization," *Expert Systems with Applications*, vol. 37, no. 12, pp. 8770–8783, 2010.

[34] G. Fang, N. M. Kwok, and Q. P. Ha, "Automatic fuzzy membership function tuning using the particle swarm optimisation," in *Proceedings of the Pacific-Asia Workshop on Computational Intelligence and Industrial Application (PACIIA '08)*, vol. 2, pp. 324–328, December 2008.

[35] G. Fang, N. M. Kwok, and D. Wang, "Automatic rule tuning of a fuzzy logic controller using particle swarm optimisation," in *Artificial Intelligence and Computational Intelligence: Part II*, vol. 6320 of *Lecture Notes in Computer Science*, pp. 326–333, 2010.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 272350, 9 pages
http://dx.doi.org/10.1155/2013/272350

Research Article

A Robust Finite Element Analysis of the Rail-Wheel Rolling Contact

Mohammad Reza Aalami,[1] Aras Anari,[1] Torkan Shafighfard,[1] and Siamak Talatahari[2]

[1] *Department of Mechanical Engineering, University of Tabriz, Tabriz 51666-14766, Iran*
[2] *Department of Civil Engineering, University of Tabriz, Tabriz 51666-14766, Iran*

Correspondence should be addressed to Siamak Talatahari; siamak.talat@gmail.com

Received 8 August 2013; Accepted 28 August 2013

Academic Editor: Amir H. Alavi

Appraisal of contact stresses, surface cracks, and plastic deformations in rails and wheels has always been an important issue in mechanical and railway engineering because of two main reasons. In the first place, these inappropriate events lead to the reduction of service life of the railway track. Besides, studying railway systems requires both time-consuming analysis methods and expensive experimental works. In this paper, a railway system containing wheel, rail, axle, and pads is modeled and analyzed. Using elastic-plastic materials, mapped meshing, and the rolling motion of the wheel contingent upon the up-to-date international railway systems results in high accuracy in the solutions of this problem. ANSYS software is utilized with the purpose of simulating the system. The contribution of this study is on the basic way of managing Rail-Wheel interaction problems from a finite element method point of view. So, stress distribution, elastic and plastic strains as well as nodal forces are considered, simultaneously. The results obtained from the simulation have suitable agreement with the real life experiences. Another feature of this paper is that it demonstrates essential steps for more realistic 3D solutions to the aforementioned problems.

1. Introduction

Railway transportation system, as one of the notable means of commuting systems, has served for human societies and has pursued its improvements as other promoted aspects of life. In recent years, the capacity of carrying axial loads for world railways as well as their velocities has been enhanced which results in increasing the amount of strains and stresses on lines and digression of rails. By this augmentation, interactions between railway components become more considerable. The rolling contact of a wheel on a rail is the basis of many Rail-Wheel related problems including the rail corrugation, wear, plastic deformation, rotating interaction fatigue, thermo-elastic-plastic behaviour in contact, fracture, creep, and vehicle dynamics vibration. Therefore, it has attracted a lot of researchers to various railway networks.

The stress distribution is an important factor at the Rail-Wheel contact interfaces, that is, two materials contacting at rolling interfaces which are extremely influenced by geometry of the contacting surfaces, material attributes, and loading and boundary conditions. Three different procedures have conventionally been utilized to inspect Rail-Wheel contacts including Hertz's analytical method [1], Kalker's programs, and contact and fastsim [2]. The calculation of these stresses becomes much more complicated in three dimensional real size geometries. For this reason, many scientists have simplified the problem mainly by means of theoretical or numerical approaches based on the Hertz's theory, which can be considered the starting point of all subsequent researches. Both static and dynamic contact stresses have been carefully examined. Convincing theories as well as computer softwares have been developed to evaluate all the influential parameters involving in the Rail-Wheel interaction [3–5]. Furthermore, in [6, 7] a closed form of analytical equations was presented to calculate the elements interaction or Hertzian stresses. For these elements, only elastic properties of materials were used; this means that elastic-plastic characteristics were ignored. Besides, they generally neglected the friction coefficient between the rail and wheel which is one of the most critical factors in determining the precise amount of stresses and

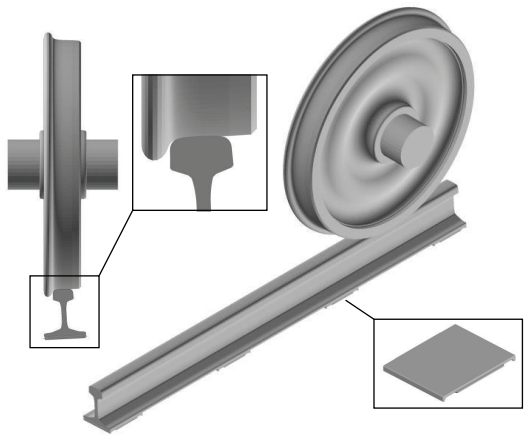


Figure 1: Generalized model of rail, wheel, axle, and pads.

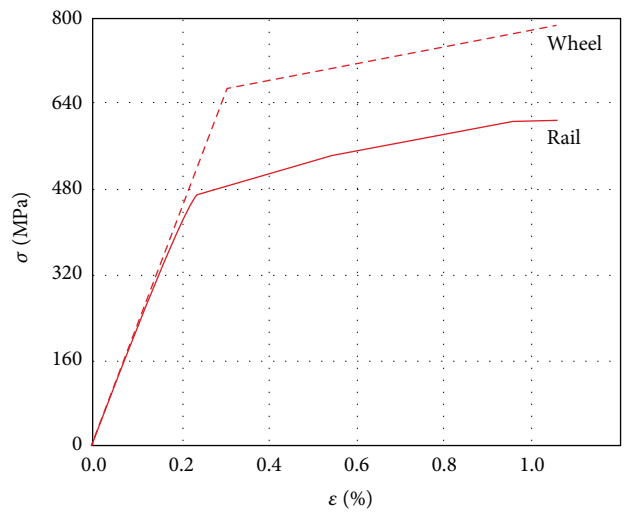


Figure 2: Stress-Strain curves for wheel and rail material models.

distribution of contact pressure in Rail-Wheel contact area. On the other hand, some practical methods have also been introduced to solve traditional problems related to Rail-Wheel interaction [8–10].

Recently, tendency towards finite element method (FEM) has increased because of its simplicity, accuracy, cost-efficiency, and versatility. FE analysis results in a set of simultaneous algebraic equations. References [11–15] have used this method in various ways with different software packages in order to obtain the optimized solutions for different Rail-Wheel interaction problems, especially stress distribution, wear, and crack growth analysis. Sladkowski and Sitarz [16] made an attempt to analyze the Rail-Wheel interaction for different profiles. Arslan and Kayabaşi [17] focused on the basic way of managing Rail-Wheel contact problems from the FEA standpoint. Guo et al. [18] utilized dynamic explicit to obtain dynamic stress and strain in the Hadfield steel crossing due to the wheel rolling passages. Bogdański et al. [19] presented a stress analysis of rail rolling contact fatigue cracks. Also, some solutions in statics have been published [20, 21]. 2D explicit solution has also been used to obtain the Rail-Wheel contact oscillatory states subjected to sliding conditions [22].

The contribution of this research is to establish a widely known FEM on a 3D Rail-Wheel, axle, and pads in order to evaluate stresses, strains, and contact forces in this system containing three main constituents (rail, wheel, and pads). However, unlike many previous works, this study focuses on the real conditions of the problem including exact boundary and loading conditions, with using real-size completed model of different components with precise profiles. To simulate the 3D contact interaction analysis of Rail-Wheel, the following stages are considered:

(i) accurate modeling of the railway system,

(ii) material modeling that is, elastic or elastic-plastic,

(iii) boundary conditions,

(iv) loading conditions,

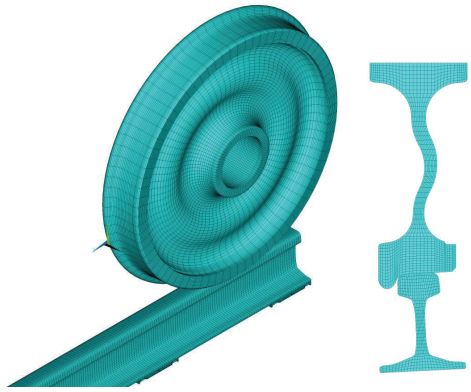


Figure 3: Mapped meshed section and extruded model.

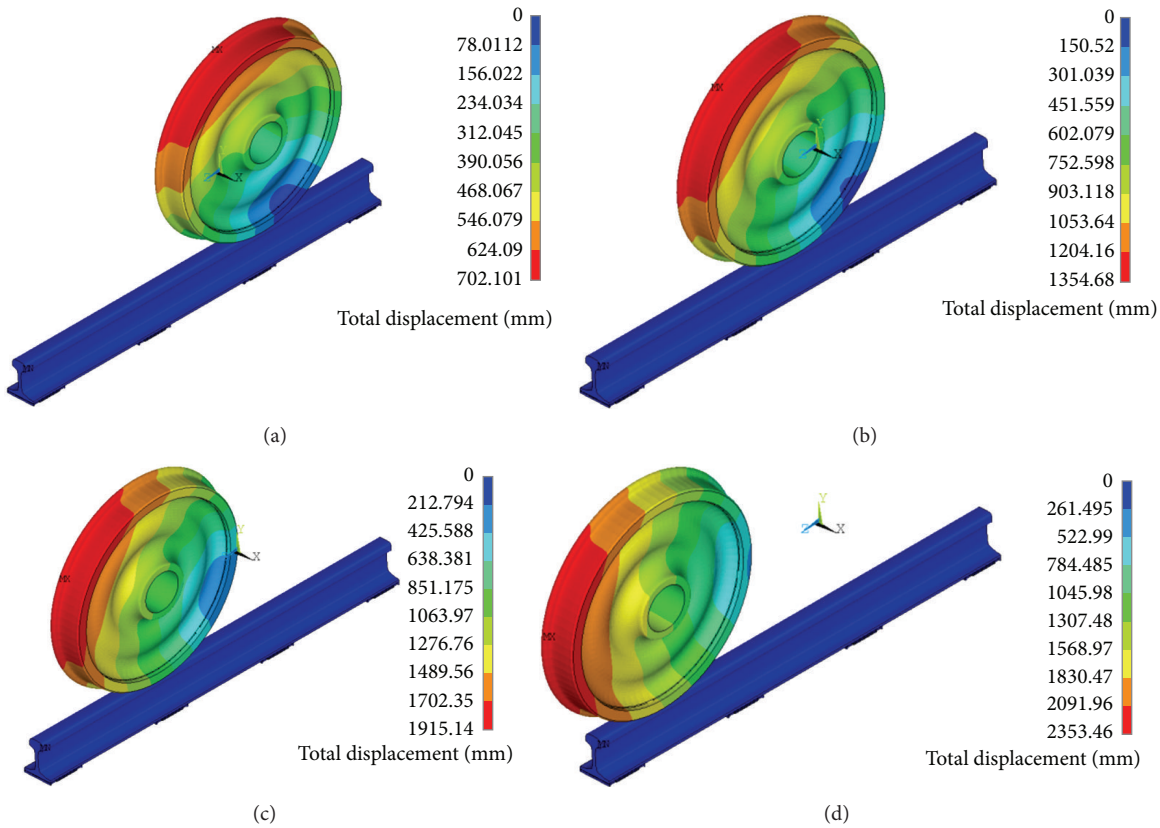


Figure 4: Total displacement of the wheel on the rail in 4 time instances (a) $t = 0.25$, (b) $t = 0.5$, (c) $t = 0.75$, and (d) $t = 1$.

(v) combination of rotational and linear motion of the wheel,

(vi) defining contacts to impacted elements,

(vii) friction coefficient,

(viii) postprocessing level.

2. Materials and Methods

2.1. Finite Element Model. To create the FE model, the precise profiles in accordance with the actual sizes have been taken into consideration. Therefore, components of this system are modeled through the accurate engineering drawings via ANSYS FE package as the analysis software. In addition, many of simplifications which reduce the accuracy of the solution are neglected. A generalized FE model of rail, wheel, axle, and pads is shown in Figure 1.

The required data to model the rail is obtained from [23]. The slope of the rail in international railways can be either 1 : 20 or 1 : 40. In this study, the slope of 1 : 20 is utilized, since it can be adjusted easily with sleepers' slope. ORE S1002 monoblock standard profile with diameter of 920 millimeter

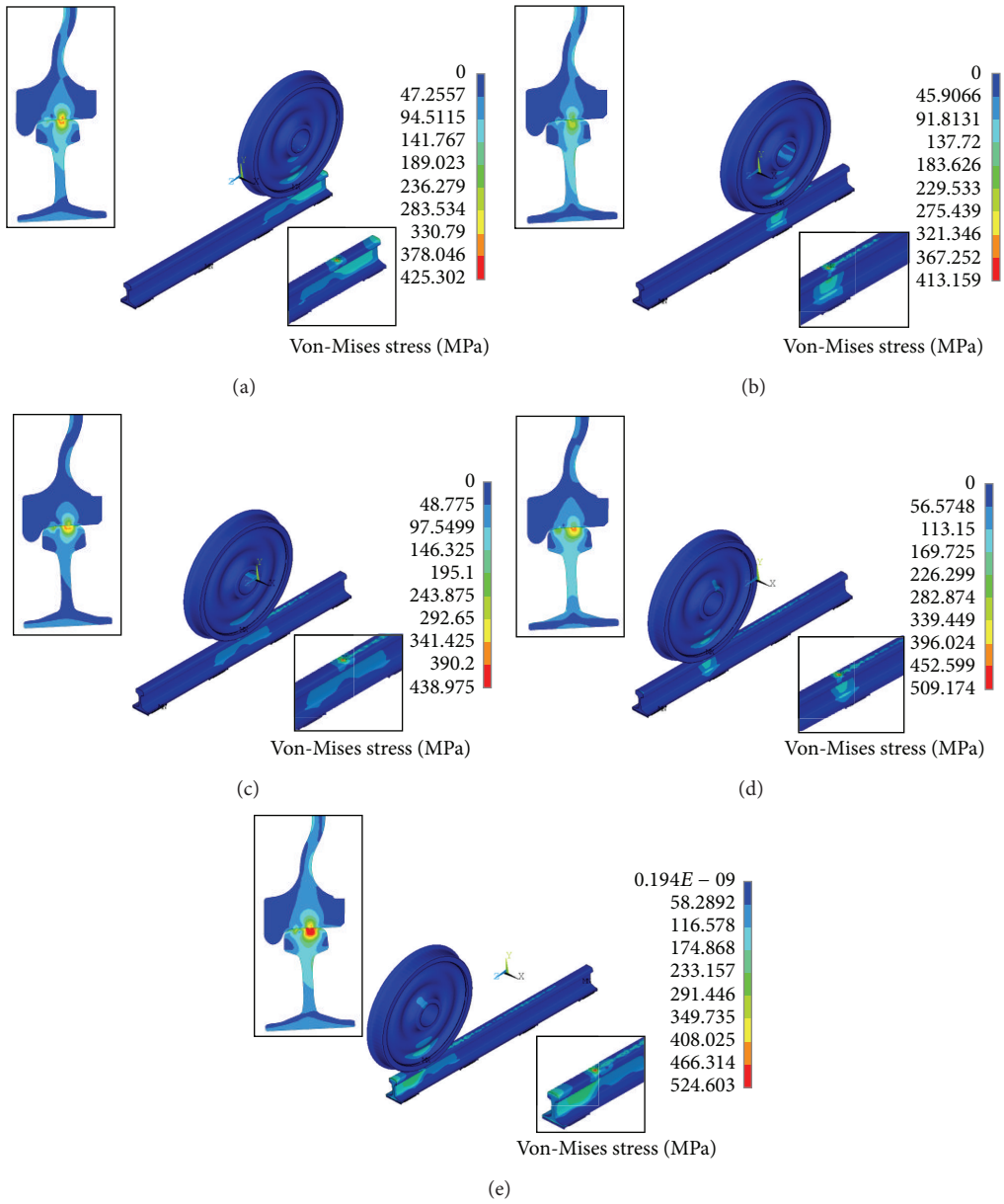


Figure 5: Von-Mises stress distribution in the system for 5 time instances (a) $t = 0.00625$, (b) $t = 0.25$, (c) $t = 0.50$, (d) $t = 0.75$, and (e) $t = 1$.

Table 1: Utilized materials for the system.

Part name	Modulus of elasticity (GPa)	Modulus of plasticity (GPa)	Yield stress (MPa)	Poisson's ratio
Rail	206.9	22.7	483	0.295
Wheel	205	22.7	640	0.3
Pad	1	—	—	0.45
Axle	205	—	—	0.3

is used for the wheel [24]. The rail's length is 1980 millimeter which includes four pads in order to make sufficient space for the rolling motion of the wheel. The plastic pads placed between the base of the steel rails and concrete sleepers are crucial in distribution of stresses, damping the axial loads applied to the sleeper, and isolation of the rail. The PAE-2 setting pads are in accordance with the UIC-60 rail, and their dimensions are complied with [25].

2.2. Material Properties. Mechanical properties for materials of the system components are shown in Table 1. Additionally,

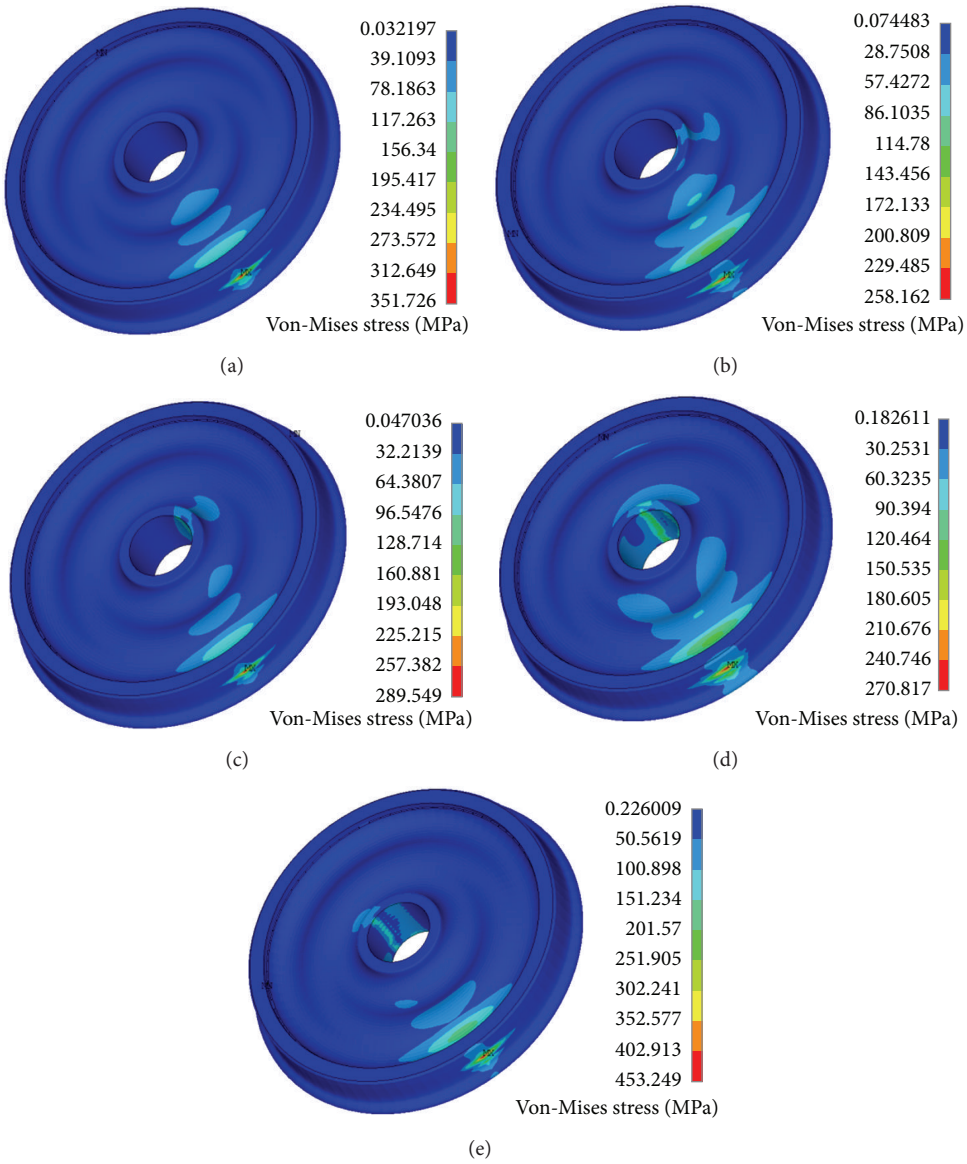


Figure 6: Von-Mises stress distribution in the wheel for 5 time instances (a) $t = 0.00625$, (b) $t = 0.25$, (c) $t = 0.50$, (d) $t = 0.75$, and (e) $t = 1$.

bilinear kinematic hardening elastic-plastic properties of the wheel and rail are shown in Figure 2 [26].

2.3. Meshing. The model is meshed with 3D, 8-Node structural SOLID185 elements [27]. Mapped mesh is utilized in order to obtain a highly precise solution with a logical model size. Sections of both the rail and the wheel are mapped meshed and extruded as depicted in Figure 3. It is worth to note that this problem is solved with different sizes of elements, and the optimum solution is prescribed by trial and error to obtain the converged solution. As a result, tiny elements are used in contact areas. A total of 217565 nodes and 262689 elements (including structural and contact elements) have been considered. Table 2 represents detailed numbers of elements and nodes for the FE model.

Table 2: Detailed numbers of the elements and nodes for the FE model.

Part	No. of elements	No. of nodes
Rail	95200	110952
Wheel	78000	86560
Axle	6900	7704
Pads	43312	12348

2.4. Interactions. Between the rail and wheel as well as between the rail and pads, a typical contact has been defined with 3D, 8-Node surface-to-surface Contact-174 and 3D Target-170 segment elements with standard behaviour. Also, the bonded contact between the axle and the wheel is characterized by 3D, 8-Node surface-to-surface Contact-174

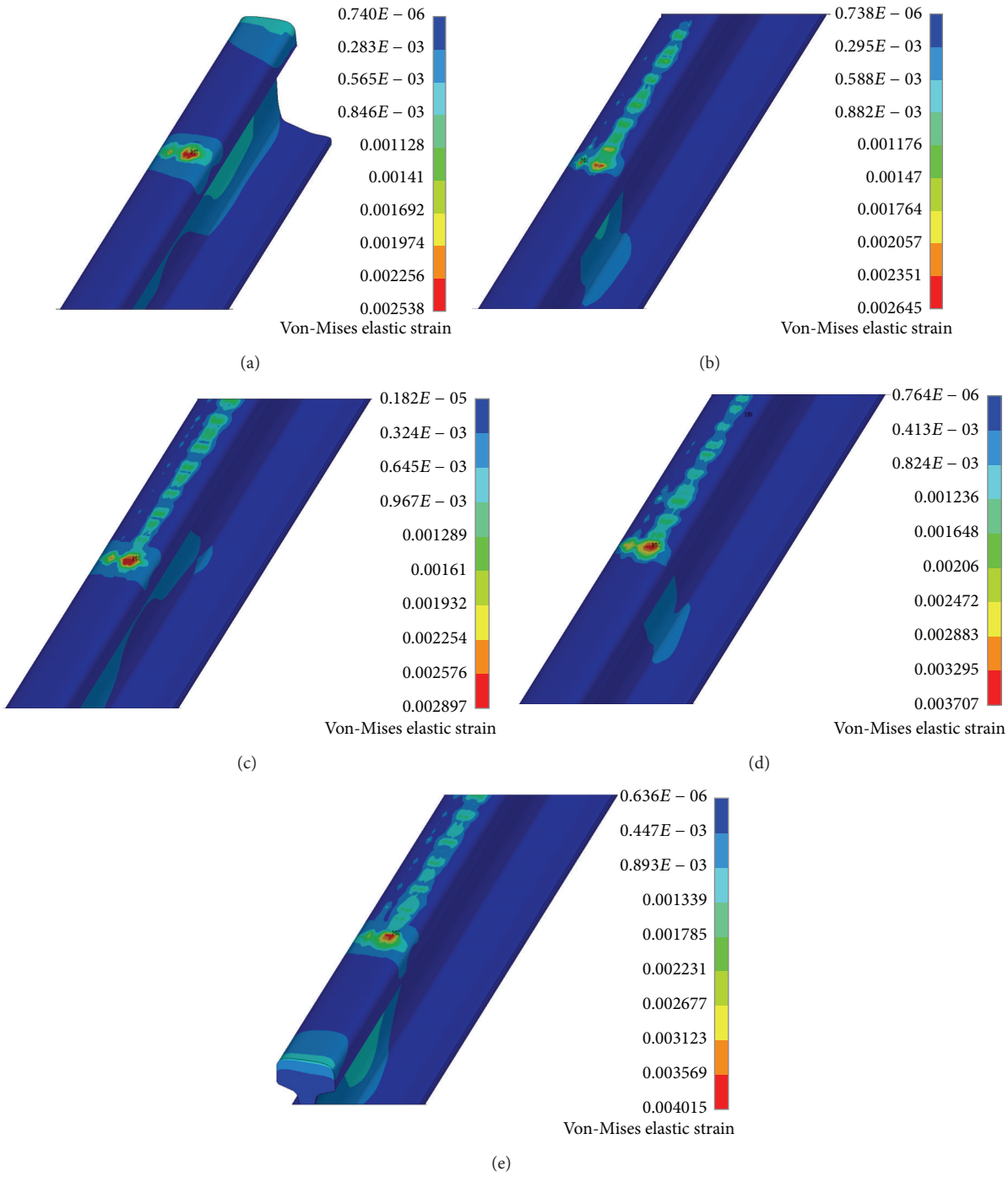


Figure 7: Von-Mises elastic strain distribution in the rail for 5 time instances (a) $t = 0.00625$, (b) $t = 0.25$, (c) $t = 0.50$, (d) $t = 0.75$, and (e) $t = 1$.

and 3D Target-170 segment. These elements have adaptation capability with SOLID185 elements and can simulate the influence of friction between the rail and wheel [27]. Here, the friction coefficient of 0.2 is chosen for the FE analysis, [28].

2.5. Boundary Conditions. In this study, the geometrical boundary conditions are applied based on physical characters and real conditions of the railway system. Symmetry boundary conditions have been implemented to both ends of the rail. Y and Z directions of pads have been constrained because they are held by sleepers, and X direction of them is restricted as they are hindered by guide plates. On the other hand, displacements along the X and Y axes as well as rotation about the Y and Z axes of the axle are constrained on account of keeping the slope of the wheel on the rail.

Due to the rolling motion of the wheel, the axle has been given the rotation of 3 radians about its axis and a 1380 mm

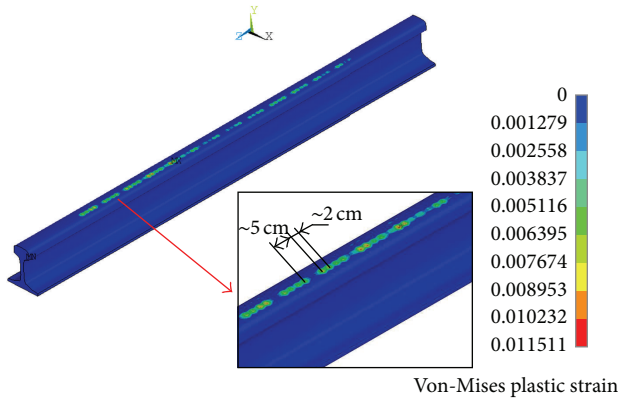


Figure 8: Von-Mises plastic strain distribution in the rail.

displacement along the rail direction which is in accordance with the approach of motion of the instantaneous center. Instead of applying vertical forces on any component, the wheel has been pushed into the rail to obtain reaction forces equal to the quasi-static loads based on [16]. Derailment coefficient has been considered in the range of safety limit, while vertical and lateral reaction forces were being calculated.

3. Results and Discussions

The analysis of Von-Mises stress in the rail and the wheel are performed and the plastic and elastic strains in Rail-Wheel contact areas as well as assessment of contact nodal normal forces are evaluated, and then, the diagram of vertical force versus solution time for the critical contact areas is carried out. In this context, the rotation of 3 radians and a 1380 millimeter displacement are applied to the center of the axle. Rotational displacement of the wheel on the rail, for 4 time instances, is exhibited in Figure 4. This displacement is given to the wheel because of two significant reasons. First, it can ensure that the route of the wheel includes two setting pads completely, which are important in assessing the nature of the contact forces fluctuated along the wheel's route. Second, this distance is required for the rolling contact to obtain the required quasi-static state.

Figure 5 shows Von-Mises stress distribution in the system as the wheel rolls on the rail, in 5 time instances. Also, the perpendicular section to the axis of the wheel route and an isometric view of the rail in contact zone are presented. The maximum amount of the stress for the rail on the Rail-Wheel contact areas is 524 MPa as illustrated in Figure 5. The amount of the stress created by the wheel motion on the rail exceeds the yield stress of the rail (483 MPa). This is because of plastic deformations, so the corrugations of the surface take place. According to Figure 6, the maximum Von-Mises stress for the wheel is 453 MPa in the last time instance, even though we cannot ignore small amount of stresses created in both the rim and the plate of the wheel. Note that the shape of the contact zone is relatively elliptic which is adjusted with the Hertzian method for the stress distribution. Besides, it should be taken into account that the rail has plastic deformations, since a large part of the contact forces between the rail and wheel is tolerated by the rail.

In every cycle of the wheel motion, the Von-Mises elastic and plastic strains are created as shown in Figures 7 and 8, respectively. The elastic strains are available above the section of the rail, as well as in the body of it. Plastic strains are merely created above the section of the rail. Hence, accumulation of these plastic strains among the time will result in the alteration of the rail's profile and its corrugation. As presented in Figure 9, corrugation with wavelengths of 4–8 cm occurs due to the plastic strains along the rail which is a result of rolling contact on the rail. Contour of the plastic strains of the rail in Figure 8 is exactly matched with the real-life experience shown in Figure 9. It illustrates that profiles of the corrugated areas are similar to zones subjected to plastic strains.

Contact status, contact penetration, and contact pressure at the end step are also exhibited in Figure 10. The maximum amount of contact pressure is obviously placed in the center of the ellipse. Contact status demonstrates approximately elliptical shape of the sliding contact zone similar to contact penetration shape.

The final part of this section is devoted to nodal forces as a fundamental part of the interpretation of other findings in FEM software packages. During the rolling motion of the wheel, the existence of pads under the rail leads to the regular oscillation of vertical nodal forces in the rail and wheel. Diagram in Figure 11 elucidates the fluctuation of nodal forces in the axle center. To obtain these results, instead of setting the vertical load and utilizing the flexibility method, mutual interaction displacement between wheel and rail is set, and the displacement approach is used. As mentioned in Section 1, the axial loads in new railway systems have been augmented; thus, the vertical force applied by the wheel in the standard Rail-Wheel interaction is chosen 200–375 kN, since it can be more than 200 kN for problems with quasi-static characteristic [16, 26].

4. Conclusions

Finite element analysis (FEA) is utilized as a tool for contact mechanics modeling, assessment, and simulation of the Rail-Wheel contact through improving the traditional

Figure 9: Real-life rail corrugation with wavelengths in the interval 4–8 cm occurs due to the accumulation of plastic strains [29].

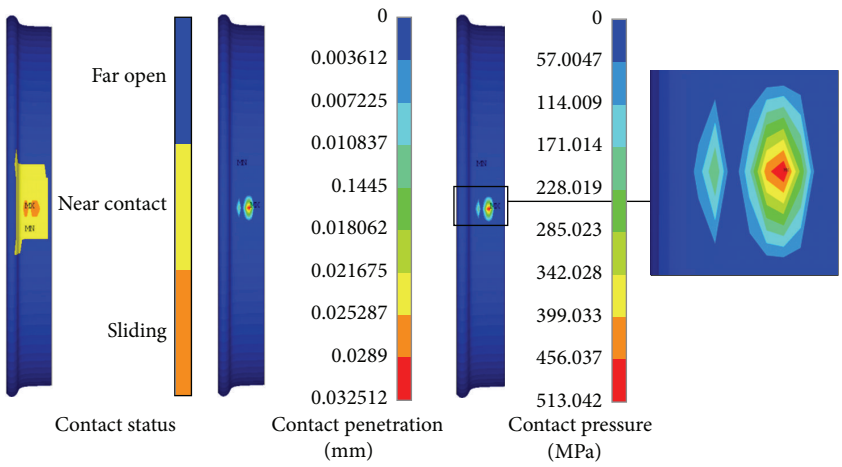


Figure 10: Contact status, penetration, and pressure distribution at the end step.

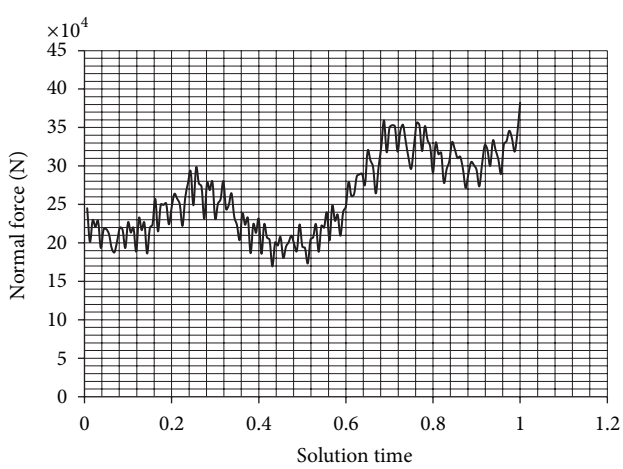


Figure 11: Variation of normal force versus solution time.

approaches of investigating the impact of the wheel motion on the rail. Most of the conventional methods and computational approaches are limited to static solutions without taking into consideration the complete rolling of the wheel. Furthermore, some simplifications of models of the railway system, that is, ignoring the setting pads in these studies, decreased the exactness of the solution. Followings are conclusions obtained by ANSYS software.

(i) The displacement applied on the shaft in vertical direction is based on the magnitude of vertical forces which are validated by both experimental and numerical analyses of previous researches.

(ii) Results show that the Von-Mises stress on the rail is above its yield point and also the shape of stress distribution is elliptical. Either of these results is verified by several numerical studies.

(iii) Resulted by rolling contact forces, plastic strains can lead to abrasion, corrugation, wear, and crack propagation. As mentioned, plastic strain zones are the same as the plastic deformations in real-life Rail-Wheel interactions.

The present study can be as one of fundamental researches for rolling contact causing fatigue, wear, and abrasion, and therefore, improving and pursuing this study can be considered in future works. Directions of the future research are to investigate the influence of pads in nodal forces, stress distribution, and strains created by the wheel on the rail. It is worth to note that pads are also subjected to different types of lateral and vertical loads. Therefore, stresses and strains on pads are needed to be studied.

References

[1] K. L. Johnson, *Contact Mechanics*, Cambridge University Press, Cambridge, UK, 1985.

[2] J. J. Kalker, *Three-Dimensional Elastic Bodies in Rolling Contact*, Kluwer Academic, Dodrecht, The Netherlands, 1990.

[3] J. J. Kalker, "Survey of wheel—rail rolling contact theory," *Vehicle System Dynamics*, vol. 8, no. 4, pp. 317–358, 1979.

[4] B. Paul, "A review of rail-wheel contact stress problems," in *Railroad Track Mechanics and Technology*, A. D. Kerr, Ed., pp. 323–352, Pergamon Press, London, UK, 1975.

[5] B. Paul, "Fundamental studies related to wheel-rail contact stress," Federal Railroad Administration Report FRA/ORD-81/05, Washington, DC, USA, 1981.

[6] R. L. Norton, *Machine Design: An Integrated Approach*, Prentice Hall, 2nd edition, 2000.

[7] J. E. Shigley and C. R. Mischke, *Mechanical Engineering Design*, Mc Graw Hill, 6th edition, 2001.

[8] D. T. Eadie, D. Elvidge, K. Oldknow et al., "The effects of top of rail friction modifier on wear and rolling contact fatigue: full-scale rail-wheel test rig evaluation, analysis and modelling," *Wear*, vol. 265, no. 9-10, pp. 1222–1230, 2008.

[9] O. Arias-Cuevas, Z. Li, R. Lewis, and E. A. Gallardo-Hernández, "Rolling-sliding laboratory tests of friction modifiers in dry and wet wheel-rail contacts," *Wear*, vol. 268, no. 2-3, pp. 543–551, 2010.

[10] A. Rovira, A. Roda, M. B. Marshall, H. Brunskill, and R. Lewis, "Experimental and numerical modelling of wheel-rail contact and wear," *Wear*, vol. 271, no. 5-6, pp. 911–924, 2011.

[11] M. Pau, F. Aymerich, and F. Ginesu, "Distribution of contact pressure in wheel-rail contact area," *Wear*, vol. 253, no. 1-2, pp. 265–274, 2002.

[12] Y. Liu, L. Liu, and S. Mahadevan, "Analysis of subsurface crack propagation under rolling contact loading in railroad wheels using FEM," *Engineering Fracture Mechanics*, vol. 74, no. 17, pp. 2659–2674, 2007.

[13] W. Daves and F. D. Fischer, "Modelling of the plastification near the rough surface of a rail by the wheel-rail contact," *Wear*, vol. 253, no. 1-2, pp. 241–246, 2002.

[14] H. Kanehara and T. Fujioka, "Measuring rail/wheel contact points of running railway vehicles," *Wear*, vol. 253, no. 1-2, pp. 275–283, 2002.

[15] M. Farjoo, W. Daniel, and P. A. Meehan, "Field statistical and finite element analysis of rail squats," *Engineering Fracture Mechanics*, vol. 109, pp. 117–129, 2012.

[16] A. Sladkowski and M. Sitarz, "Analysis of wheel-rail interaction using FE software," *Wear*, vol. 258, no. 7-8, pp. 1217–1223, 2005.

[17] M. A. Arslan and O. Kayabaşi, "3-D rail-wheel contact analysis using FEA," *Advances in Engineering Software*, vol. 45, no. 1, pp. 325–331, 2012.

[18] S. L. Guo, D. Y. Sun, F. C. Zhang, X. Y. Feng, and L. H. Qian, "Damage of a Hadfield steel crossing due to wheel rolling 8 impact passages," *Wear*, vol. 305, no. 1-2, pp. 267–273, 2013.

[19] S. Bogdański, M. Olzak, and J. Stupnicki, "Numerical stress analysis of rail rolling contact fatigue cracks," *Wear*, vol. 191, no. 1-2, pp. 14–24, 1996.

[20] M. Bijak-Zochowski and P. Marek, "Residual stress in some elasto-plastic problems of rolling contact with friction," *International Journal of Mechanical Sciences*, vol. 39, no. 1, pp. 15–32, 1997.

[21] J. Zhang, S. Sun, and X. Jin, "Numerical simulation of two-point contact between wheel and rail," *Acta Mechanica Solida Sinica*, vol. 22, no. 4, pp. 352–359, 2009.

[22] A. Saulot and L. Baillet, "Dynamic finite element simulations for understanding wheel-rail contact oscillatory states occurring under sliding conditions," *Journal of Tribology*, vol. 128, no. 4, pp. 761–770, 2006.

[23] *UIC Leaflet 861-3: Standard 60 Kg/M Rail Profiles-Types UIC 60 and 60E*, The International Union of Railways, 3rd edition, 2002.

[24] *UIC Leaflet 510-2: Conditions Concerning the Use of Wheels of Various Diameters*, The International Union of Railways, 4th edition, 2004.

[25] I. A. Carrascal, J. A. Casado, J. A. Polanco, and F. Gutiérrez-Solana, "Dynamic behaviour of railway fastening setting pads," *Engineering Failure Analysis*, vol. 14, no. 2, pp. 364–373, 2007.

[26] U. Sellgren, T. Telliskivi, U. Olofsson, and P. Kruse, "A tool and a method for FE analysis of wheel and rail interaction," in *Proceedings of the international ANSYS conference*, Pittsburgh, Pa, USA, 2000.

[27] "ANSYS," http://www.ansys.com.

[28] X. Zhao and Z. Li, "The solution of frictional wheel-rail rolling contact with a 3D transient finite element model: validation and error analysis," *Wear*, vol. 271, no. 1-2, pp. 444–452, 2011.

[29] J. C. O. Nielsen, "High-frequency vertical wheel-rail contact forces-validation of a prediction model by field testing," *Wear*, vol. 265, no. 9-10, pp. 1465–1471, 2008.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 124903, 13 pages
http://dx.doi.org/10.1155/2013/124903

Research Article

An Improved Genetic-Simulated Annealing Algorithm Based on a Hormone Modulation Mechanism for a Flexible Flow-Shop Scheduling Problem

Min Dai,[1,2] Dunbing Tang,[1,2] Kun Zheng,[1,2] and Qixiang Cai[1,2]

[1] *College of Mechanical and Electrical Engineering, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China*
[2] *Jiangsu Key Laboratory of Precision and Micro-Manufacturing Technology, Nanjing 210016, China*

Correspondence should be addressed to Dunbing Tang; d.tang@nuaa.edu.cn

Received 22 April 2013; Revised 10 July 2013; Accepted 14 July 2013

Academic Editor: Shengyong Chen

A flexible flow-shop scheduling (FFS) with nonidentical parallel machines for minimizing the maximum completion time or makespan is a well-known combinational problem. Since the problem is known to be strongly NP-hard, optimization can either be the subject of optimization approaches or be implemented for some approximated cases. In this paper, an improved genetic-simulated annealing algorithm (IGAA), which combines genetic algorithm (GA) based on an encoding matrix with simulated annealing algorithm (SAA) based on a hormone modulation mechanism, is proposed to achieve the optimal or near-optimal solution. The novel hybrid algorithm tries to overcome the local optimum and further to explore the solution space. To evaluate the performance of IGAA, computational experiments are conducted and compared with results generated by different algorithms. Experimental results clearly demonstrate that the improved metaheuristic algorithm performs considerably well in terms of solution quality, and it outperforms several other algorithms.

1. Introduction

A flexible or hybrid flow-shop scheduling problem is a further development of the classical flow-shop scheduling [1]. FFS problems have been widely applied in process industries and flexible manufacturing systems, such as iron and steel, printed circuit board, textile, and chemical industries [2]. In the literature, most of the studies related to FFS problems are concentrated on multiple stages involved with identical parallel machines which have the same processing capacity [3–6]. However, due to newer or more modern facilities running side by side with older and less efficient ones, it is common that practical production is executed on nonidentical parallel machines in a real manufacturing system. In other words, the processing time differs in unrelated parallel machines for the same operation. In this paper, the FFS with nonidentical parallel machines is considered that is, there are different parallel machines at each production stage, and each job can be processed with different speeds at one production stage.

Scheduling problems is considered to be NP-hard in a flexible flow shop. It has been proved that the FFS with two production stages, where one machine is in the first production stage and several machines are in the second production stage, is NP-hard [7]. Therefore, many variants of algorithms are employed to solve such problems. In terms of the algorithms characteristics, there are three categories for solving FFS problems, that is, exact methods, heuristic, and metaheuristic algorithms [8]. Exact methods that have focused on simplified versions of FFS problems can precisely obtain best solutions using mathematical models. Branch and bound is the preferred approach for solving the problems. A detailed survey for branch and bound approach is firstly proposed to investigate FFS problems with the objective of minimizing the makespan by Salvador [9]. Later, various branch and bound algorithms for flow-shop problems with parallel machines have been defined [10–13]. These exact algorithms can guarantee optimal solutions in small-size problems. However, a major disadvantage of the optimization

approaches to production scheduling is the need to reach the better solution quality with respect to large-scale scheduling problems.

On the other hand, when FFS problems grow in complexity and data volume, a variety of approximate methods are usually employed to explore solution qualities, that is, heuristic and metaheuristic algorithms. These algorithms are used to solve such problems and to get optimal or near-optimal solutions with considerably less running time. Dispatching rules are one of the most outstanding paradigms of heuristic algorithms. For instance Sriskandarajah and Sethi [14] presented heuristic algorithms based on dispatching rules for a flexible flowshop problem with the minimum makespan criterion. Verma and Dessouky [15] studied a multistage problem with identical jobs and uniform parallel machines to minimize the makespan and investigated the performance of dispatching rules. In further study, researchers have developed prominent strategies to enhance the performance of heuristics, known as metaheuristic algorithms. Metaheuristics like genetic algorithm (GA) and simulated annealing algorithm (SAA) have been successfully used in FFS problems. Oĝuz and Ercan [16] described the GA with a new crossover operator to solve the hybrid flow-shop scheduling with multiprocessor task problems. Kahraman et al. [17] developed an efficient GA for hybrid flow-shop scheduling problems with the objective of minimizing the makespan. Wang et al. [18] presented the SAA to study a hybrid flow-shop scheduling problem with multiprocessor tasks under the makespan criterion. Mirsanei et al. [19] provided a novel SAA algorithm with a new neighborhood function to obtain a better result of the makespan in a hybrid flow-shop scheduling problem with identical parallel machines. Other metaheuristic algorithms have also been applied to FFS, such as ant colony optimization (ACO) [20], neural network (NN) [21], and particle swarm optimization (PSO) [22]. In addition, combinational metaheuristic algorithm which absorbs advantages of more than one algorithm is one of the most important approaches. The researches that highlight simulated annealing algorithms are integrated together with genetic algorithms, which have provided promising results in scheduling problems [23–25]. Most of these researches demonstrate that heuristic and metaheuristic approaches have been performed to reduce the gap between the theories and their practical production scheduling, but they have not yet been solved satisfactorily.

To the best of our knowledge, the research on combinational metaheuristics of GA and SAA for FFS problems has not been investigated. In making a step towards reducing the gap between the theory and their practice, this paper focuses on the development and performance of an improved genetic-simulated annealing algorithm (IGAA) for minimizing the makespan in a flexible flow shop with non-identical parallel machines. Due to the FFS problem with NP-hard, an IGAA with a new crossover and mutation operation is developed to obtain optimal or near-optimal solutions of the makespan. More specifically, the crossover operation that is based on an encoding matrix is designed to create legal schedules and to ensure the population diversity.

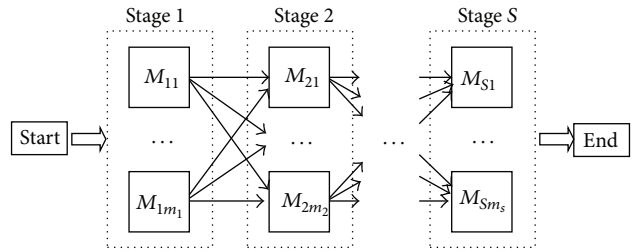


Figure 1: A flexible flow-shop layout.

The mutation operation that is based on an improved SAA can avoid falling into a local optimum.

The remainder of this paper is organized as follows. In Section 2, an FFS problem is described, and a linear mixed integer programming model with the objective of minimizing the makespan during the flexible flow-shop environment is constructed. In Section 3, a metaheuristic approach IGAA for solving the scheduling optimization problem is used. In Section 4, computational experiments are conducted to test the performance of IGAA for FFS problems. In Section 5, the conclusions are concluded and future work is introduced.

2. The Mathematical Model for an FFS Problem

2.1. The Problem Description. The FFS is a multistage production process that is composed of two or more production stages in series. There is at least one machine tool at each production stage, and at least one stage has more than one machine tool. All jobs have to go through every production stage in the same order. The FFS has infinite intermediate storage between any successive production stages [26]. One instance of the FFS consists of a set of J jobs and a set of M machine tools. Each job j ($j \in J$) on machine i ($i \in M$) has corresponding processing time at a given speed. All jobs are available to be processed sequentially and nonpreemptively at different machine stages as illustrated in Figure 1. The scheduling objective of the FFS is to assign jobs to machine tools at the corresponding stages and determine the processing sequence of operations on each machine in order to minimize the maximum completion time, that is, the makespan ($C_{\max}$).

The constraints of FFS are made as follows.

(1) One job can be processed by only one machine at each production stage.

(2) One machine can process at most one operation at a time.

(3) For the first stage, all jobs are available at time $t = 0$.

(4) There are no precedence relationships between operations of different jobs, but there are precedence relationships between different operations of one job.

(5) Preemption is not allowed for processing each job that is, once an operation is started, it must be finished without interruption.

(6) Every operation of one job can be executed in a machine at a given speed for every stage.

(7) For the same operation, the processing time differs in different unrelated parallel machines in a production stage.

2.2. The Mathematical Model. The following parameters and decision variables are associated with the FFS problem:

(i) S is the set of spindle speeds for one machine tool;

(ii) J is the set of jobs;

(iii) s is the set of production stages that all jobs have to be executed;

(iv) M_l is the set of machine tools at production stage $l (l \in s)$;

(v) L is a very large positive number;

(vi) T_{jlmv} $(j \in J, l \in s, m \in M_l, v \in S)$ is the processing time when job j at production stage l is processed on machine tool m with speed v;

(vii) $S_{jlm} (j \in J, l \in s, m \in M_l)$ is the starting time when job j at production stage l is processed on machine tool m;

(viii) C_{jlm} $(j \in J, l \in s, m \in M_l)$ is the completion time when job j at production stage l is processed on machine tool m;

(ix) $C_{\max}$ is the makespan of the schedule, that is, the completion time of the last job in the schedule;

(x) T_D is due date;

(xi) $X_{jlmv} (j \in J, l \in s, m \in M_l, v \in S)$ is an integer variable that have two possible values: 0 or 1, it is set to 1 if job j at production stage l is required to process on machine tool m with speed v, and it is set to 0 otherwise;

(xii) Y_{jklm} $(j, k \in J, j \neq k, l \in s, m \in M_l)$ is an integer variable that have two possible values: 0 or 1, it is set to 1 if job j is before job k at production stage l on machine tool m, and it is set to 0 otherwise.

The following is a linear mixed integer programming model that discovers a production sequence for jobs on the machines in order to minimize the makespan:

$$\text{minimize} \quad C_{\max} \qquad \text{subject to} \tag{1}$$

$$C_{jlm} \leq C_{\max}, \quad j \in J;\ l \in s;\ m \in M_l \tag{2}$$

$$C_{\max} \leq T_D \tag{3}$$

$$\sum_{m \in M_l} X_{jlmv} = 1, \quad j \in J;\ l \in s;\ v \in S \tag{4}$$

$$\sum_{v \in S} X_{jlmv} = 1, \quad j \in J;\ l \in s;\ m \in M_l \tag{5}$$

$$C_{jlm} = S_{jlm} + \sum_{v \in S} T_{jlmv} X_{jlmv}, \quad j \in J;\ l \in s;\ m \in M_l \tag{6}$$

$$C_{jlm} \leq S_{j(l+1)n}, \quad j \in J;\ l \in s,\ l \neq s;\ m \in M_l;\ n \in M_{l+1} \tag{7}$$

$$C_{jlm} \leq S_{klm} + L\left(1 - Y_{jklm}\right), \quad j, k \in J,\ j \neq k;\ l \in s;\ m \in M_l. \tag{8}$$

Constraints (2)-(3) define that the makespan, which requires arrival before the due date, is equal to the completion time of the last job in the schedule. Constraint (4) means that one job can be assigned to only one machine tool at each production stage; that is, it does not allow a job to be executed on more than one machine tool at any time. Constraint (5) imposes that one job can be processed on one machine tool with one chosen speed. Constraint (6) points out that the completion time of job j is composed of the processing time and starting time at each production stage. Constraint (7) gives the precedence constraints between the operations of job j; that is, one operation of the job cannot be processed at next production stage until it has been finished at the current stage. Constraint (8) ensures that one machine can process next job only after it has finished the current one; that is, it does not allow more than one job to be executed on a machine tool at the same time.

3. IGAA Design for an FFS

In this section, an improved hybrid metaheuristic algorithm is proposed for a flexible flow-shop scheduling problem. There are many metaheuristic algorithms that have been implemented in an FFS, such as genetic algorithm (GA), simulated annealing algorithm (SAA), particle swam optimization (PSO), and ant colony optimization (ACO). Among these approaches, GA can quickly approach the optimization solution, but a fatal shortcoming is that it is liable to be trapped in a local optimum, that is, premature convergence. Fortunately, SAA has the ability to jump out of the local optimization and search for the best solution. Therefore,

Table 1: A 3 × 3 flexible flow-shop scheduling problem.

Job number	Process 1			Process 2		Process 3	
	Machine (M1)	Machine (M2)	Machine (M3)	Machine (M4)	Machine (M5)	Machine (M6)	Machine (M7)
1	2	2	3	4	5	7	6
2	6	5	4	3	4	4	7
3	3	5	4	6	5	3	2

this paper proposes to incorporate the strengths of a simulated annealing algorithm into a genetic algorithm. GA is developed to rapidly search for an optimal or near-optimal solution among the solution space, and then SAA is utilized to seek a better one on the base of the solution. Furthermore, due to the low search efficiency of the SAA, a novel annealing rate function, which is inspired from hormone modulation mechanism, is adopted to further improve the efficiency of the exploration. The proposed improved genetic-simulated annealing algorithm (IGAA) for an FFS is illustrated in Figure 2.

3.1. The Matrix Encoding and Decoding Representation. In general, each chromosome that consists of a series of genes is a bit string structure using binary coding or real-value coding. The length of the chromosome has a direct influence on genetic operation and running time. The longer the length is, the more complex genetic operation become, and the longer running time is. In terms of the elements and their corresponding positions in a matrix describing the constraints between jobs, a matrix encoding approach for an FFS is presented. In this representation, the encoding matrix is described as a whole, instead of a bit string structure generated by the matrix. One dimensional chromosome is converted into a multidimensional chromosome, which is defined as a matrix chromosome. The advantage of the matrix-chromosome is convenient to select, cross, and mutate. Moreover, it can ensure the completeness and validity of the offspring. Hence, each matrix-chromosome represents a legal and feasible schedule. Suppose that N jobs are to be processed on a set of machine tools and each job is required to pass S stages. There are M_s $(s = 1, 2, \ldots, \mathrm{S})$ unrelated parallel machine tools at each production stage. A matrix-chromosome $N \times S$ is constructed as follows:

$$A_{N\times S} = \begin{bmatrix} a(1,1) & a(1,2) & \cdots & a(1,S) \\ a(2,1) & a(2,2) & \cdots & a(2,S) \\ \vdots & \vdots & a(i,j) & \vdots \\ a(N,1) & a(N,2) & \cdots & a(N,S) \end{bmatrix}. \quad (9)$$

The matrix elements $a(i, j)$: $(a(i, j) \in (1, M_s + 1)$, $i = 1, 2, \ldots, N$, $j = 1, 2, \ldots, S$, $s = 1, 2, \ldots, S)$ are random real numbers. $\mathrm{Int}(a(i, j))$ is the integer of $a(i, j)$; it indicates the machine tools' identifier that deals with the jth process of job i. In decoding step, if the condition $\mathrm{Int}(a(h, j)) = \mathrm{Int}(a(i, j))$ and $h \neq i$ is satisfied, then it means that there are several jobs waiting to be processed on the same machine tool for the same process. When the process is the first one, these jobs are arranged to operate in accordance with the ascending sequence of $a(i, 1) : (i = 1, 2, \ldots, N)$. When the process number is greater than one, these jobs are determined by their completion time of previous process. In other words, the shorter the finishing time of previous process is, the earlier the next process can be operated. If the completion time is the same, jobs are operated according to the ascending sequence of $a(i, j)$, $(i = 1, 2, \ldots, N$, $j = 2, 3, \ldots, S)$. For example, assume that 3 jobs are scheduled at 3 production stages in a flexible flow shop. Each job has 3 processes. The number of parallel machines for each stage is 3, 2, and 2. The example of a 3 × 3 flexible flow-shop scheduling problem is given in Table 1.

A matrix-chromosome based on the encoding rule is generated randomly using Matlab simulation software as follows:

$$A = \begin{bmatrix} 1.1316 & 2.3234 & 2.1123 \\ 3.4342 & 1.1453 & 2.4357 \\ 3.9015 & 2.6456 & 1.2137 \end{bmatrix}. \quad (10)$$

The matrix (10) is converted into the matrix (11) in the form of the integer as expressed in the following:

$$A^* = \begin{bmatrix} 1 & 2 & 2 \\ 3 & 1 & 2 \\ 3 & 2 & 1 \end{bmatrix}. \quad (11)$$

According to the matrix (10), the relationship between jobs and machine tools can be described as in the following.

(i) Three processes of job 1 can be processed on M1, M5, and M7, respectively.

(ii) Three processes of job 2 can be processed on M3, M4, and M7, respectively.

(iii) Three processes of job 3 can be processed on M3, M5, and M6, respectively.

And then on the base of the rule of the matrix decoding, the sequence of jobs on the same machine tool can be obtained:

(i) M1: the first process of job 1,

(ii) M2: readiness,

(iii) M3: the first process of job 2 → the first process of job 3 (Due to 3.9015 > 3.4342),

(iv) M4: the second process of job 2,

(v) M5: the second process of job 1 → the second process of job 3 (Due to 2.6456 > 2.3234),

(vi) M6: the third process of job 3,

(vii) M7: the third process of job 1 → the third process of job 2 (Due to 2.4357 > 2.1123).

The matrix encoding and decoding scheme along with the Gantt chart for a possible solution is depicted in Figure 3.

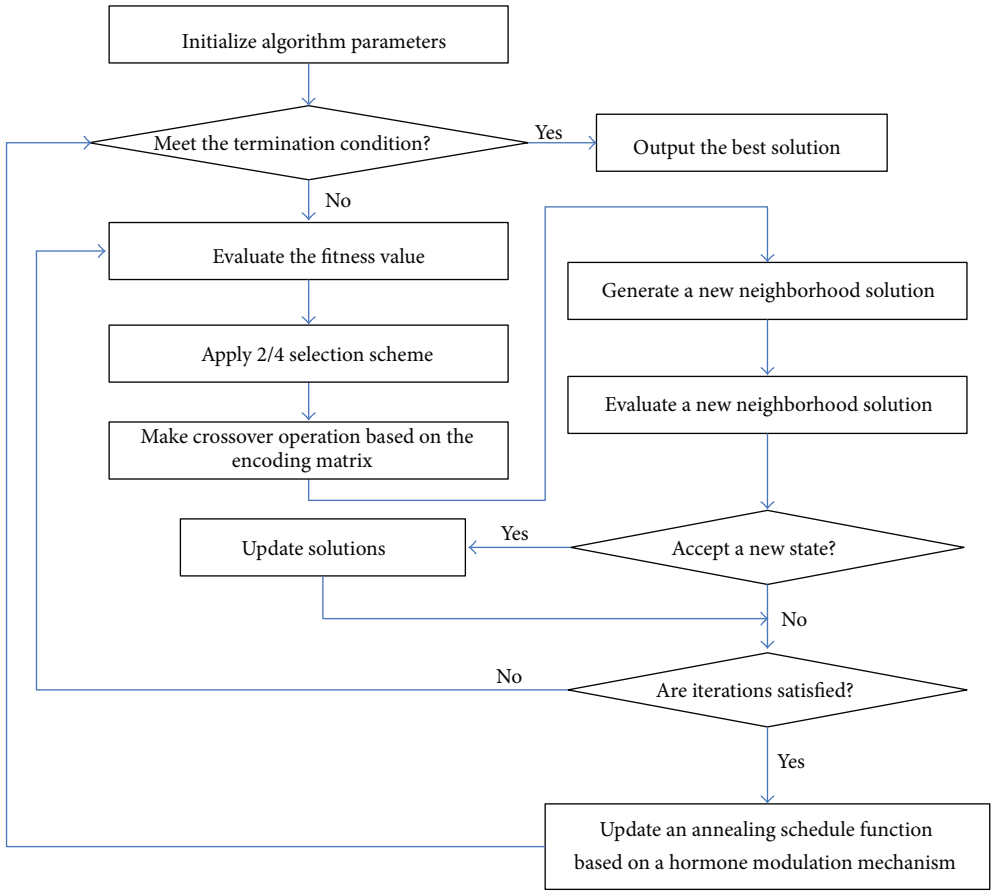


Figure 2: The flow chart of an improved genetic-simulated annealing algorithm.

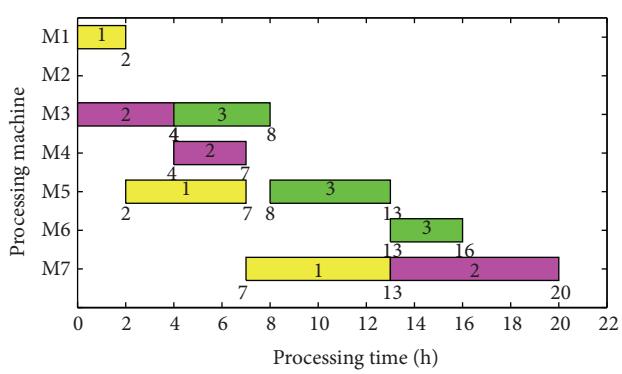


Figure 3: The Gantt chart of the flexible flow-shop scheduling (3 × 3).

3.2. Fitness Function. The genetic-simulated annealing algorithm assesses the solution that is based on the fitness function. The greater fitness an individual has, the higher chance it has to be chosen into the next generation. In general, the fitness is related with the objective function. As the objective function is to minimize the makespan, the aforementioned function can be transformed into the fitness function for solution k as follows:

$$F(k) = \frac{1}{C_{\max}(k)}. \tag{12}$$

3.3. Selection Scheme. Selection operator has a significant impact on the performance of the GA. On the base of the fitness of the matrix-chromosome, the selection operator chooses matrix-chromosomes that are used for crossover and mutation. Often the fitness value is not the best one. In the literature, some remarkable selection schemes which have been applied to scheduling problems are developed to explore good solution space. A "2/4 selection" is adopted to preserve fittest matrix chromosomes at each generation, and maintain the diversity of the matrix population as well [27].

$$\text{Parent A} = \left[\begin{array}{c|ccc|cc} a(1,1) & a(1,2) & \cdots & a(1,j) & \cdots & a(1,S) \\ a(2,1) & a(2,2) & \cdots & a(2,j) & \cdots & a(2,S) \\ \vdots & \vdots & & \vdots & & \vdots \\ a(N,1) & a(N,2) & \cdots & a(N,j) & \cdots & a(N,S) \end{array}\right]$$

$$\text{Parent B} = \left[\begin{array}{c|ccc|cc} b(1,1) & b(1,2) & \cdots & b(1,j) & \cdots & b(1,S) \\ b(2,1) & b(2,2) & \cdots & b(2,j) & \cdots & b(2,S) \\ \vdots & \vdots & & \vdots & & \\ b(N,1) & b(N,2) & \cdots & b(N,j) & \cdots & b(N,S) \end{array}\right]$$

$$\text{Child A'} = \left[\begin{array}{c|ccc|cc} a(1,1) & b(1,2) & \cdots & b(1,j) & \cdots & a(1,S) \\ a(2,1) & b(2,2) & \cdots & b(2,j) & \cdots & a(2,S) \\ \vdots & \vdots & & \vdots & & \vdots \\ a(N,1) & b(N,2) & \cdots & b(N,j) & \cdots & a(N,S) \end{array}\right]$$

$$\text{Child B'} = \left[\begin{array}{c|ccc|cc} b(1,1) & a(1,2) & \cdots & a(1,j) & \cdots & b(1,S) \\ b(2,1) & a(2,2) & \cdots & a(2,j) & \cdots & b(2,S) \\ \vdots & \vdots & & \vdots & & \vdots \\ b(N,1) & a(N,2) & \cdots & a(N,j) & \cdots & b(N,S) \end{array}\right]$$

Figure 4: Two-column crossover operator (TCO).

3.4. Crossover Operator Based on an Encoding Matrix. Crossover operator is the main way to create new population. A crossover operation picks randomly two parents with crossover probability (P_c) [28]. In other words, if P_c is greater than a uniformly distributed random number generated on the interval 0 to 1, the crossover operation can be applied to the parents. A large number of crossover operator approaches, such as one-point crossover, two-point crossover, multi point crossover, uniform crossover, and nonuniform crossover have been reported widely in the literature. In this study, based on two-point crossover technique a novel crossover operator, namely, two-row/column crossover operator (TCO), for our proposed heuristic algorithm is introduced. TCO can be viewed as an extension of two-point crossover operation. The aim of developing TCO can not only always produce feasible permutations, but also effectively increase the diversity of population and further enhance the ability to explore solution space. Only if each matrix-chromosome element satisfies the encoding condition $a(i,j) \in (1, M_s + 1)$, $(i = 1, 2, \ldots, N;\ j = 2, 3, \ldots, S;\ s = 1, 2, \ldots, S)$, TCO will yield legal offspring and will not need a repairing procedure. A permutation representation of TCO with P_c can be described in the following.

A column permutation operation is considered between two matrix-chromosomes, that is, Parent A and Parent B. Two different cut-off columns are randomly picked with P_c between the range 1 and $S - 1$. New matrix-chromosomes are generated by relocating zones between the cut-off columns of paired matrix chromosomes. The codes between 2th and

jth column of matrix-chromosomes given in Figure 4 are changed.

3.5. Mutation Operator Based on an Improved SAA Operation. As crossover operation cannot yield solutions with new information, it requires mutation operation with some probability for each segment in order to generate the solutions with greater fitness. A mutation operator is divided into two important issues. One is the proportion of population implementing mutation, that is, mutation probability (P_m). If P_m is greater than a random number generated on the interval 0 to 1 using uniformly distributed rule, the mutation operation can be executed. The other is the mutational strength, that is, the perturbation produced in a matrix-chromosome. There are a variety of approaches for mutation operation, such as uniform mutation and nonuniform mutation [29], power mutation [30], and mutation operator based on the immunity operation [31].

As the general mutation operator is inefficient and insufficient for solving complex global optimization problems, a mutation operation based on an improved SAA operation is proposed for the IGAA. When a temperature is high enough, the search space is very enormous and the SAA can accept a new state with a quite large probability; when a temperature is low, the search space becomes very small and the SAA can accept a new state with a small probability. Thus, the new mutation operation can enhance the search ability and search efficiency of the IGAA in the solution space.

In Figure 2 some parameters with respect to the SAA are required to investigate the performance of the IGAA, including the initial temperature, the neighborhood structure, the annealing rate, and the termination condition. These factors play a significant role in the performance of the IGAA and should be implemented carefully as follows.

(1) The initial temperature. The initial temperature T_0 should be set to a high enough value to ensure a vast number of accepted states at an initial stage. It decreases slowly during the iterations of the algorithm. As the temperature becomes lower, it is likely that the state will be accepted with a small probability. It is possible that the IGAA can avoid falling into a local optimum. Finally, a global optimum solution or near-optimal solution will be obtained. Note that the initial temperature value should be set appropriately. Setting a too high temperature value will consume computation time largely, while setting a too low one will reject many accepted states. In this paper, the initial temperature function in the IGAA can be set as

$$T_0 = \frac{-100 \times \left(U_{\max} - U_{\min}\right)}{\log \mathrm{Pr}}, \tag{13}$$

where $U_{\max}$ is the maximum sum of the processing time of all jobs, $U_{\min}$ is the minimum sum of the processing time of all jobs, and Pr is the initial acceptance probability.

(2) The neighborhood structure. Neighborhood structures have a direct impact on the efficiency of local search. A neighborhood structure can yield a new set of neighbor solutions by changing the order sequence of the current operation for a given solution. One of the most effective neighborhood strategies for the SAA regarding a production scheduling problem is based on the critical path method [32]. One critical path which is decomposed into a number of blocks corresponds to one feasible solution. One block represents a maximal sequence of several operations that either are required to process on the same machine tool or belongs to the same job. In this study, moving an operation of one critical block to the end of the block or the beginning of the block is adopted to generate the neighborhood.

(3) The annealing rate function. The performance of the IGAA has a significant relation with an annealing rate [33]. In order to enhance the search efficiency of the IGAA, a novel annealing rate method, which is inspired from a hormone modulation mechanism, is developed. Farhy [34] pointed out that the hormone modulation had characteristics with monotone and nonnegative. A control function with nonlinearity is employed, that is, the upregulatory or downregulatory Hill function, as shown in the following:

$$F_{\mathrm{up}}(\mathrm{X}) = \frac{X^n}{T^n + \mathrm{X}^n} \quad \text{or} \quad F_{\mathrm{down}}(X) = \frac{T^n}{T^n + X^n}, \tag{14}$$

where T is a threshold, $T > 0$; X is an independent variable; n is a Hill coefficient, $n \geq 1$. Note that $F_{\mathrm{up}} + F_{\mathrm{down}} = 1$ and $0 \leq F_{\mathrm{up}}(X) \leq 1$, $0 \leq F_{\mathrm{down}}(X) \leq 1$. The Hill functions can reach a quick stability, which keeps hormone modulation adaptive and stable. If one hormone a is controlled by another hormone b, the secretion of the former V_a is determined by the concentration of the latter C_b, which can be described as

$$V_a = c_0 F\left(C_b\right) + V_{a0}, \tag{15}$$

where V_{a0} is the basal secretion of hormone a, and c_0 is a constant.

Based on the previous hormone modulation mechanism, the annealing rate function can be designed as follows:

$$\begin{aligned} T(k+1) &= \alpha * F_{\mathrm{down}}(k) - \frac{k * \Delta T}{\exp(k)} \\ &\text{subject to} \\ F_{\mathrm{down}}(k) &= \frac{1}{(1+k^n)}, \\ \Delta T &= T(k+1) - T(k), \end{aligned} \tag{16}$$

where α is a small constant, k is the number of iterations, and n is a Hill coefficient, $n \geq 1$. ΔT is the difference between the current temperature $T(k+1)$ and the previous temperature $T(k)$, and $\Delta T < 0$. The annealing rate function is exemplified in the plots in Figure 5 (for $n = 1, 1.2, 1.5$, and 2 and $k = 25$).

(4) The terminating condition. In the SAA, a terminating criterion which is used to end the annealing procedure consists of the Markov chain stability criterion and the external circulation stopping criterion. In this study, calculating the iterations at each given temperature decides if the condition of the Markov chain stability criterion is satisfied or not. When no improvement in the fitness values can be obtained, the IGAA will be ended.

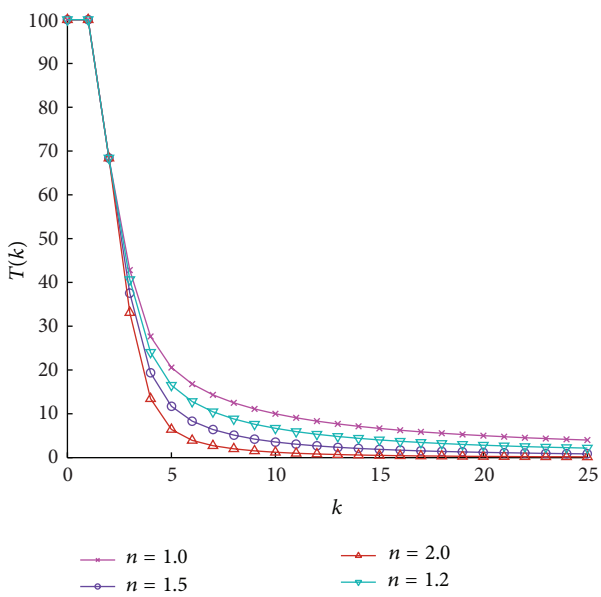


Figure 5: Function profiles of the annealing rate (for $n = 1$, $n = 1.2$, $n = 1.5$, and $n = 2$).

4. Computational Experiments

In this section, the improved genetic-simulated annealing algorithm is used to analyze the flexible flow-shop scheduling problem with the objective of minimizing the makespan. To evaluate the performance of the algorithm, two experiments are conducted. Test problems consist of the number of jobs, number of machines, and number of production stages and range of processing times. The simulation is carried out by utilizing Matlab programming language. The experimental tests are displayed on a personal computer with Intel Pentium (R) with 1 GB memory and 3.20 GHz processor, and its operating system is Windows XP.

4.1. Evaluation of IGAA. In the first experiment the performance of IGAA is evaluated through using an average relative percentage deviation (ARPD) with respect to the optimal solution. ARPD is obtained by the given formula in the following:

$$\text{ARPD} = \frac{\sum_{i=1}^{I} \text{PD}(i)}{I}, \tag{17}$$

where I represents the number of replications for each instance. PD which is used as the performance of the makespan represents the percentage deviation of one algorithm. The percentage deviation (PD) is defined as follows:

$$\text{PD} = \frac{C_{\max} - S_{\text{best}}}{S_{\text{best}}} \times 100, \tag{18}$$

where $C_{\max}$ denotes the makespan value produced by the algorithm and S_{best} is corresponding to the best result of objective function in the replications.

The choice of algorithm parameters may affect solution quality. First, two important parameters regarding IGAA are considered in this paper. One is the crossover probability (P_c). The other is the mutation operation which is replaced with improved simulated annealing algorithm. The principle of incorporating the improved SAA based on the hormone modulation mechanism into the genetic algorithm is the most central innovation of this paper. Therefore, computational experiments are conducted to illustrate that different factors of annealing rate functions really have significant influence on the quality of the solutions. The population size is fixed at 30; annealing rate function factor (n): 1.0, 1.2, 1.5, and 2.0; crossover probability (P_c): 0.5, 0.6, 0.7, 0.8, and 0.9; jobs (N): 20 and 50; production stages (S): 2, 5, and 8; the processing times for each job at each production stage are uniformly yielded over the interval from 1 to 100. The algorithm was run 10 times for each instance. The computational results of different parameters are shown in Table 2. The minimum values of ARPD with different crossover probability (P_c) for all problem sizes are in bold and with different annealing rate functions factor (n) in italic.

In Table 2, the annealing rate function factor $n = 1.5$ is the best when crossover probability is fixed. Meanwhile, the best crossover probability is $P_c = 0.9$ when the annealing rate function factor is fixed. Therefore, the annealing rate function factor with 1.5 and the crossover probability with 0.9 are used to evaluate the performance of the proposed algorithm in the following experiments.

In addition, the characteristics used to describe the problem in the proposed algorithm, such as the population size of jobs, the number of production stages, and the number of machines at each production stage, are required to assess the performance of solution quality. Three experiments are conducted for different sets of these characteristics as displayed in Figures 6, 7, and 8, respectively. It is given that the processing times of jobs are uniformly ranged from 1 to 100.

In Figure 6 an experiment is conducted for a 50 jobs ×5 stages problem in which there are five machine tools at each production stage. The population size varies from 10 to 30. As shown in Figure 6, it is obvious that the IGAA obtains the optimal or near-optimal solution quickly when the number of population size increases.

In Figure 7 the number of jobs is 50, the number of production stage varies from 3 to 5, and the number of machine tools at each production stage is fixed at 5. The population size is given at 25. It is observed from Figure 7 that the performance of the IGAA with less production stages performs better.

In Figure 8 an experiment was conducted with a five-stage and 50-job instance. The number of machines at each production stage varies from 3 to 7. The population size is given at 25. The graph clearly shows that increasing the number of machine tools is beneficial to improve the algorithm performance, and thereby more machine tools are selected as the proposed algorithm.

4.2. Comparison with Several Existing Algorithms. To evaluate the solution quality of the IGAA, the algorithm is compared with several different algorithms in the second

Table 2: The effect of different parameters on the performance of IGAA.

Problem size $N \times S$	Function factor n	$P_c = 0.5$	$P_c = 0.6$	$P_c = 0.7$	$P_c = 0.8$	$P_c = 0.9$
20×2	1.0	11.43	10.90	10.73	9.82	**6.57**
	1.2	*10.71*	10.71	10.22	9.57	**7.29**
	1.5	12.14	*10.00*	*9.29*	*7.57*	***3.71***
	2.0	14.29	12.79	11.56	10.21	**7.29**
20×5	1.0	10.23	10.09	9.83	9.15	**8.71**
	1.2	13.64	11.02	9.69	7.19	**5.30**
	1.5	*9.85*	*8.21*	*6.34*	*5.00*	***4.17***
	2.0	12.50	11.35	11.00	10.39	**9.70**
20×8	1.0	17.91	14.79	13.13	10.06	**8.54**
	1.2	14.17	11.57	10.40	9.13	**5.63**
	1.5	*12.92*	*10.67*	*8.05*	*7.11*	***4.67***
	2.0	19.04	16.15	13.37	10.17	**7.54**
50×2	1.0	18.64	15.77	12.24	11.31	**6.71**
	1.2	*17.26*	14.29	10.19	8.73	**5.19**
	1.5	17.56	*12.11*	*10.00*	*7.35*	***4.87***
	2.0	19.34	16.06	14.15	11.70	**8.54**
50×5	1.0	20.74	15.74	12.86	10.67	**8.52**
	1.2	20.37	15.00	13.09	11.24	**7.78**
	1.5	*15.19*	*12.19*	*11.13*	*8.58*	***5.04***
	2.0	18.52	16.67	14.81	11.71	**9.37**
50×8	1.0	22.23	17.54	15.07	11.60	**9.54**
	1.2	20.78	16.14	14.70	10.69	**8.54**
	1.5	*18.85*	*15.34*	*11.20*	*9.12*	***5.63***
	2.0	19.75	15.79	13.69	10.11	**7.97**

experiment. The compared algorithms include genetic algorithm (GA) given by Şerifoğlu and Ulusoy [35] and simulated annealing (SA) by Wang et al. [18]. The problem sizes are involved in three different numbers of jobs (i.e., $N = 20$, 60, and 100) and three different numbers of production stages (i.e., $S = 5, 10, 15$). The processing times for each job at each production stage are generated randomly from a discrete uniform distribution over the range 10 to 100. The algorithm was run five times for each instance. The experimental results of performance comparison among four algorithms with respect to the best value, the worst value, and the average value of the objective function are depicted in Table 3. The aforementioned *ARPD* used in the problem sizes is also suitable. The results of the *ARPD* for GA, SA, and IGAA are presented in Table 4.

In Table 3, the best values of objective function are displayed in boldface. It is observed that IGAA is an effective algorithm for obtaining the optimal or near-optimal solution of all instances. The IGAA provides the better chance to further improve the quality of the solutions during the 37 instances of the total 45 instances. All algorithms obtain optimal solution for the remaining instances, but it only focuses on small-scale and medium-scale problems. At the same time, it is noticed that the proposed algorithm outperforms GA and SA with respect to the worst value and the average value of the objective function.

In Table 4, the total average ARPD of IGAA is 1.467 while the total average ARPD for GA and SA are 4.066 and 3.104, respectively. It is obvious that IGAA is superior to GA and SA in solution quality. Furthermore, the improvement rate for the total average ARPD using IGAA is defined in the following:

$$\begin{aligned}&\text{Improvement Rate (\%)}\\&\quad = (\text{Total Average ARPD of GA or SA}\\&\qquad - \text{Total Average ARPD of IGAA})\\&\qquad \times (\text{Total Average ARPD of GA or SA})^{-1}.\end{aligned} \tag{19}$$

On the total average *ARPD* of IGAA, an improvement of 63.92% with respect to GA and 52.74% with respect to SA has been achieved.

5. Conclusions

The purpose of this study tries to make a step towards constructing an effective metaheuristic algorithm for a flexible flow-shop scheduling problem with nonidentical parallel machines to minimize the makespan. An improved genetic-simulated annealing algorithm (IGAA) is proposed. Specially,

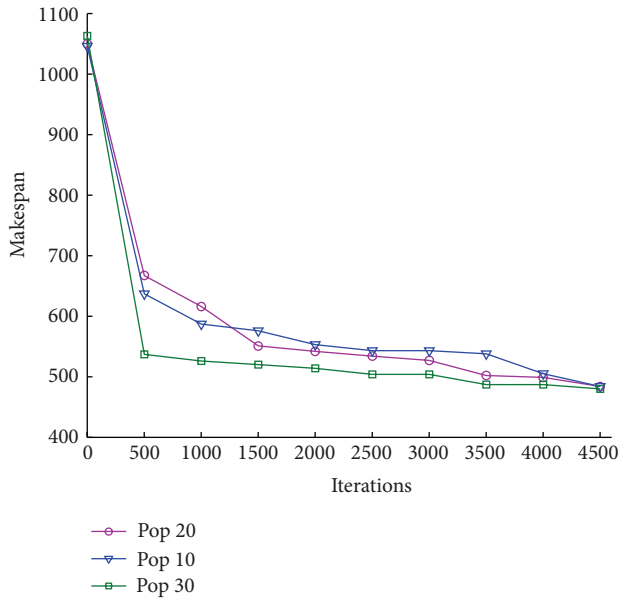


Figure 6: The performance of the IGAA with different population size.

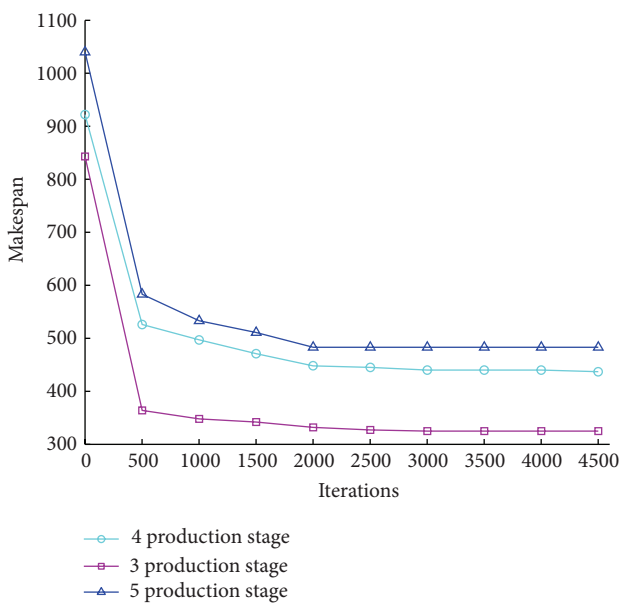


Figure 7: The performance of the IGAA with different production stages.

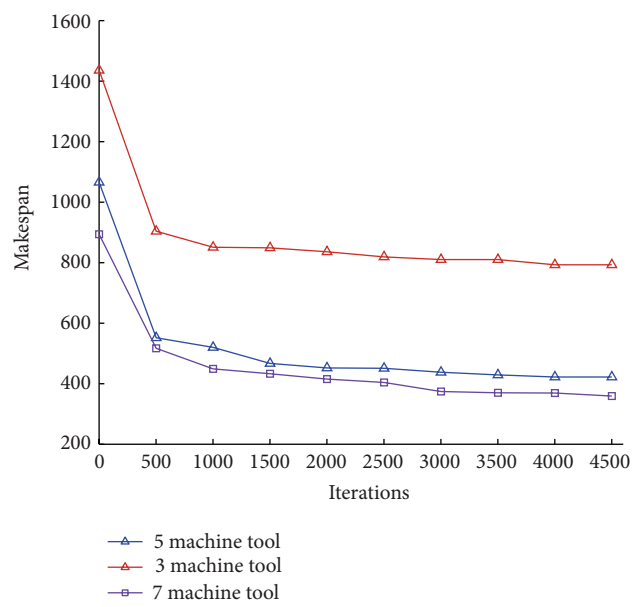


Figure 8: The performance of the IGAA with different machine tools.

a matrix encoding scheme which employs a real-value representation is developed to deal with the FFS. Next, instead of the mutation operator of GA, an improved SAA inspired from a hormone modulation mechanism is implemented into GA to avoid premature convergence. Several annealing rate functions are compared to determine the appropriate variant of IGAA for the scheduling problem. Finally, the hybrid metaheuristic algorithm is provided to solve the FFS problem.

Computational experiments are carried out to verify the performance of the IGAA with different parameters. The experimental results demonstrate that the IGAA performs better according to the average relative percentage deviation (ARPD) of the solution when the annealing rate function factor with 1.5 and the crossover probability with 0.9 are selected. At the same time, the proposed algorithm with the two appointed parameters is implemented to test other parameters. The results show the performance of the IGAA performs well when the population size of jobs, the number of production stages, and the number of machines at each production stage increases. It is furthermore observed that the makespan can reduce effectively.

Table 3: Experimental results of IGAA compared with GA and SA.

Problem size $N \times S$	GA			SA			IGAA		
	Best	Worst	Avg	Best	Worst	Avg	Best	Worst	Avg
20×5	**604**	707	647.40	607	772	669.96	**604**	**604**	**604**
	613	717	649.24	604	715	653.23	**598**	**598**	**598**
	620	729	649.40	**614**	790	689.06	**614**	**633**	**614.76**
	612	717	656.96	613	771	664.84	**612**	**612**	**612**
	613	691	634.08	**607**	711	670.24	**607**	**633**	**630.12**
20×10	1020	1103	1048.5	**1012**	1208	1149.8	1026	**1026**	**1026**
	1036	1160	1075.8	1036	1308	1284	**1001**	**1009**	**1001.3**
	1036	1128	1064.3	1026	1212	1108.5	**994**	**994**	**994**
	1058	1235	1097.8	1054	1256	1124	**1042**	**1042**	**1042**
	1064	1232	1105.2	1026	1300	1214.6	**1025**	**1028**	**1025.2**
20×15	1470	1662	1499.1	1460	1680	1524	**1430**	**1430**	**1430**
	1457	1617	1488	**1456**	1700	1598.2	**1456**	**1456**	**1456**
	1471	1647	1503.6	1459	1711	1599	**1424**	**1424**	**1424**
	1488	1625	1504.4	1457	1706	1600.1	**1454**	**1454**	**1454**
	1452	1574	1474	1471	1645	1515.7	**1443**	**1443**	**1443**
60×5	1354	1465	1319.6	**1298**	1488	1327.2	**1298**	**1298**	**1298**
	1315	1428	1347	1311	1501	1402.5	**1291**	**1291**	**1291**
	1298	1452	1332.6	1326	1579	1433	1311	**1317**	**1311.2**
	1351	1480	1377.2	1307	1602	1457.8	**1293**	**1294**	**1293.1**
	1327	1463	1358.2	1348	1611	1505.1	**1307**	**1307**	**1307**
60×10	1991	2121	2012.5	1927	2250	2100.7	**1897**	**1934**	**1881.2**
	1980	2243	2019.5	1949	2173	2078.6	**1904**	**1904**	**1904**
	1949	2077	1974.6	1935	2206	2065.4	**1916**	**1916**	**1916**
	1894	1957	1918.9	1897	2109	2012	**1887**	**1899**	**1887.5**
	1940	2033	1949.5	1940	2232	2110.4	**1929**	**1931**	**1930.0**
60×15	2499	2818	2529.8	2487	2901	2611.4	**2479**	**2479**	**2479**
	2529	2719	2605.6	2501	2913	2656.2	**2498**	**2498**	**2498**
	2465	2715	2503.6	2457	2818	2600	**2417**	**2417**	**2417**
	2516	2865	2566.9	2498	2824	2617.2	**2438**	**2438**	**2438**
	2522	2731	2561	2512	2899	2660.8	**2473**	**2536**	**2479.2**
100×5	2038	2205	2060.7	2001	2300	2104.3	**1950**	**1950**	**1950**
	2017	2133	2041.3	1999	2208	2098	**1919**	**1919**	**1919**
	2018	2140	2054.6	2012	2212	2100.5	**1943**	**1950**	**1943.3**
	2056	2134	2085.9	2034	2302	2178.6	**1974**	**2050**	**1977**
	2008	2117	2036.4	1977	2219	2112.1	**1935**	**1935**	**1935**
100×10	2752	2931	2799.6	2701	2990	2876.3	**2690**	**2717**	**2711**
	2710	2882	2752.8	2713	2919	2834	**2607**	**2620**	**2607.5**
	2804	2993	2830.8	2811	3001	2890.4	**2642**	**2687**	**2643.8**
	2705	2826	2733.1	2711	3023	2900.8	**2656**	**2656**	**2656**
	2729	2985	2776.5	2699	2920	2817	**2641**	**2653**	**2642.4**
100×15	3429	3712	3452.2	3401	3801	3516.4	**3298**	**3298**	**3298**
	3496	3637	3515	3390	3758	3512	**3350**	**3350**	**3350**
	3463	3735	3516	3412	3734	3520.9	**3405**	**3405**	**3405**
	3516	3685	3541.1	3420	3812	3601.2	**3377**	**3377**	**3377**
	3485	3710	3507.4	3411	3856	3620	**3340**	**3343**	**3342**

Table 4: ARPD of IGAA compared with GA and SA.

Problem size $N \times S$	ARPD		
	GA	SA	IGAA
20×5	2.408	1.839	1.505
20×10	4.909	3.702	2.374
20×15	3.062	2.570	1.222
60×5	2.943	2.091	0.687
60×10	3.381	2.258	1.039
60×15	3.691	3.062	1.820
100×5	5.649	4.461	1.313
100×10	5.102	4.603	1.542
100×15	5.452	3.299	1.698
Average	4.066	3.104	1.467

The experiments also involve a comparison of the IGAA with other algorithms from the literature to evaluate the quality of solutions. The results reveal that the performance of the IGAA outperforms several other algorithms. From these, we can conclude that the IGAA is an effective method for the FFS problem. Further research may investigate the problems with other constraint conditions, such as release dates and setup times.

Conflict of Interests

The authors have declared no conflict of interests.

Acknowledgments

This project is supported by National Natural Science Foundation of China (Grant no. 51175262), Jiangsu Province Science Foundation for Excellent Youths (Grant no. BK2012 10111), Jiangsu Province Industry-Academy-Research Grant (Grant no. BY201220116), NUAA Fundamental Research Funds (no. NS2013053), and Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

References

[1] R. Linn and W. Zhang, "Hybrid flow shop scheduling: a survey," *Computers & Industrial Engineering*, vol. 37, no. 1, pp. 57–61, 1999.

[2] O. Engin, G. Ceran, and M. K. Yilmaz, "An efficient genetic algorithm for hybrid flow shop scheduling with multiprocessor task problems," *Applied Soft Computing*, vol. 11, no. 3, pp. 3056–3065, 2011.

[3] B.-C. Choi and K. Lee, "Two-stage proportionate flexible flow shop to minimize the makespan," *Journal of Combinatorial Optimization*, vol. 25, no. 1, pp. 123–134, 2013.

[4] J. N. D. Gupta and A. J. Ruiz-Torres, "Minimizing makespan subject to minimum total flow-time on identical parallel machines," *European Journal of Operational Research*, vol. 125, no. 2, pp. 370–380, 2000.

[5] J. N. D. Gupta, K. Krüger, V. Lauff, F. Werner, and Y. N. Sotskov, "Heuristics for hybrid flow shops with controllable processing times and assignable due dates," *Computers & Operations Research*, vol. 29, no. 10, pp. 1417–1439, 2002.

[6] G. J. Kyparisis and C. Koulamas, "Flexible flow shop scheduling with uniform parallel machines," *European Journal of Operational Research*, vol. 168, no. 3, pp. 985–997, 2006.

[7] J. N. D. Gupta, "Two-stage, hybrid flowshop scheduling problem," *Journal of the Operational Research Society*, vol. 39, no. 4, pp. 359–364, 1988.

[8] R. Ruiz and J. A. Vázquez-Rodríguez, "The hybrid flow shop scheduling problem," *European Journal of Operational Research*, vol. 205, no. 1, pp. 1–18, 2010.

[9] M. S. Salvador, "A solution to a special case of flow shop scheduling problems," in *Symposium of the Theory of Scheduling and Applications*, S. E. Elmaghraby, Ed., pp. 83–91, Springer, New York, NY, USA, 1973.

[10] O. Moursli and Y. Pochet, "A branch-and-bound algorithm for the hybrid flowshop," *International Journal of Production Economics*, vol. 64, no. 1–3, pp. 113–125, 2000.

[11] J. N. D. Gupta and E. A. Tunc, "Schedules for a two-stage hybrid flowshop with parallel machines at the second stage," *International Journal of Production Research*, vol. 29, no. 7, pp. 1489–1502, 1991.

[12] S. A. Brah and J. L. Hunsucker, "Branch and bound algorithm for the flow shop with multiple processors," *European Journal of Operational Research*, vol. 51, no. 1, pp. 88–99, 1991.

[13] J. N. D. Gupta, A. M. A. Hariri, and C. N. Potts, "Scheduling a two-stage hybrid flow shop with parallel machines at the first stage," *Annals of Operations Research*, vol. 69, pp. 171–191, 1997.

[14] C. Sriskandarajah and S. P. Sethi, "Scheduling algorithms for flexible flowshops: worst and average case performance," *European Journal of Operational Research*, vol. 43, no. 2, pp. 143–160, 1989.

[15] S. Verma and M. Dessouky, "Multistage hybrid flowshop scheduling with identical jobs and uniform parallel machines," *Journal of Scheduling*, vol. 2, no. 3, pp. 135–150, 1999.

[16] C. Oğuz and M. F. Ercan, "A genetic algorithm for hybrid flow-shop scheduling with multiprocessor tasks," *Journal of Scheduling*, vol. 8, no. 4, pp. 323–351, 2005.

[17] C. Kahraman, O. Engin, I. Kaya, and M. K. Yilmaz, "An application of effective genetic algorithms for solving hybrid flow shop scheduling problems," *International Journal of Computational Intelligence Systems*, vol. 1, no. 2, pp. 134–147, 2008.

[18] H.-M. Wang, F.-D. Chou, and F.-C. Wu, "A simulated annealing for hybrid flow shop scheduling with multiprocessor tasks to minimize makespan," *The International Journal of Advanced Manufacturing Technology*, vol. 53, no. 5–8, pp. 761–776, 2011.

[19] H. S. Mirsanei, M. Zandieh, M. J. Moayed, and M. R. Khabbazi, "A simulated annealing algorithm approach to hybrid flow shop scheduling with sequence-dependent setup times," *Journal of Intelligent Manufacturing*, vol. 22, no. 6, pp. 965–978, 2011.

[20] K. Alaykýran, O. Engin, and A. Döyen, "Using ant colony optimization to solve hybrid flow shop scheduling problems," *The International Journal of Advanced Manufacturing Technology*, vol. 35, no. 5-6, pp. 541–550, 2007.

[21] L. Tang, W. Liu, and J. Liu, "A neural network model and algorithm for the hybrid flow shop scheduling problem in a dynamic environment," *Journal of Intelligent Manufacturing*, vol. 16, no. 3, pp. 361–370, 2005.

[22] M. R. Singh and S. S. Mahapatra, "A swarm optimization approach for flexible flow shop scheduling with multiprocessor tasks," *The International Journal of Advanced Manufacturing Technology*, vol. 62, no. 1–4, pp. 267–277, 2012.

[23] L. Wang and D. Z. Zheng, "A modified genetic algorithm for job shop scheduling," *The International Journal of Advanced Manufacturing Technology*, vol. 20, no. 1, pp. 72–76, 2002.

[24] A. C. Nearchou, "A novel metaheuristic approach for the flow shop scheduling problem," *Engineering Applications of Artificial Intelligence*, vol. 17, no. 3, pp. 289–300, 2004.

[25] A. C. Nearchou, "Flow-shop sequencing using hybrid simulated annealing," *Journal of Intelligent Manufacturing*, vol. 15, no. 3, pp. 317–328, 2004.

[26] M. Pindo, *Scheduling: Theory, Algorithms, and Systems*, Prentice Hall, Englewood Cliffs, NJ, USA, 2002.

[27] G. Shi, "A genetic algorithm applied to a classic job-shop scheduling problem," *International Journal of Systems Science*, vol. 28, no. 1, pp. 25–32, 1997.

[28] D. S. Sepich, D. C. Myers, R. Short, J. Topczewski, F. Marlow, and L. Solnica-Krezel, "Role of the zebrafish trilobite locus in gastrulation movements of convergence and extension," *Genesis*, vol. 27, no. 4, pp. 159–173, 2000.

[29] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, New York, NY, USA, 1992.

[30] K. Deep and M. Thakur, "A new mutation operator for real coded genetic algorithms," *Applied Mathematics and Computation*, vol. 193, no. 1, pp. 211–230, 2007.

[31] L.-N. Xing, Y.-W. Chen, and K.-W. Yang, "A novel mutation operator based on the immunity operation," *European Journal of Operational Research*, vol. 197, no. 2, pp. 830–833, 2009.

[32] C.-F. Liaw, "A tabu search algorithm for the open shop scheduling problem," *Computers & Operations Research*, vol. 26, no. 2, pp. 109–126, 1999.

[33] M. M. Keikha, "Improved simulated annealing using momentum terms," in *Proceedings of the 2nd International Conference on Intelligent Systems, Modelling and Simulation (ISMS '11)*, pp. 44–48, Phnom Penh, Cambodia, January 2011.

[34] L. S. Farhy, "Modeling of oscillations in endocrine networks with feedback," in *Numerical Computer Methods*, p. 54, 2004.

[35] F. S. Şerifoğlu and G. Ulusoy, "Multiprocessor task scheduling in multistage hybrid flow-shops: a genetic algorithm approach," *Journal of the Operational Research Society*, vol. 55, no. 5, pp. 504–512, 2004.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 401846, 5 pages
http://dx.doi.org/10.1155/2013/401846

Research Article

A Novel Mechanical Component Retrieval Approach Based on Differential Moment

Kehua Guo, Wu Liu, and Hong Song

School of Information Science & Engineering, Central South University, Changsha 410083, China

Correspondence should be addressed to Hong Song; songhong@csu.edu.cn

Received 24 May 2013; Accepted 24 June 2013

Academic Editor: Shengyong Chen

The differential geometry and moment invariants are important approaches in pattern recognition and artificial intelligence. In this paper, a novel approach for mechanical component retrieval is reported. This approach combines local (spatial curvature) signatures and global (moment) features of 3D mechanical component. Mean curvatures are integrated into the computation of 3D moment invariants, and differential moment invariants are constructed. Experiments indicate that differential moment invariants have a lower computation complexity than traditional 3D image retrieval approaches without reducing the retrieval rate.

1. Introduction

In mechanical engineering, searching suitable components or models from 3D image database is an important issue. This problem generally stems from many applications such as 3D shape matching [1], 3D model [2], and automatic assembly for components [3]. In the recent decades, mechanical component retrieval attracts considerable attention.

Traditional 3D models retrieval approaches can be divided into three categories: text-based image retrieval (TBIR) approach, content-based image retrieval (CBIR) approach, and semantic-based image retrieval (SBIR) [4] approach. In the three approaches, TBIR method can only support the retrieval based on keywords; SBIR approach relies on the annotation by users. Therefore, in most mechanical component retrieval systems, CBIR approach can be more simple and effective. Some typical algorithms are listed as follows.

Local Representations. In local approaches, shapes are broken down into boundary segments called primitives. These representations differ in the selection of primitives and the organization of the primitives for shape representation. Common methods include shape distributions approaches [5] and curvature-based descriptors [6, 7]. Local approaches supply precise descriptions. However, they have low robustness to noise and are complex to be implemented.

Global Representations. Global approaches compute a multidimensional numeric signature vector from shape information. Typical methods include 3D moment invariants [8, 9], 3D Fourier descriptors [10], and neural network approach [11]. Global descriptors are robust to noise and have a lower computation complexity, but they cannot achieve a perfect classification rate in recognition to mechanical components with slight different shape characteristics. Figure 1 shows some mechanical components that traditional global approach cannot distinguish.

Therefore, how to fully use the advantages of local and global features has been an important issue [12]. To solve this problem, this paper reports a novel mechanical component retrieval approach. The novel approach, named differential moment invariants, is based on two innovations. The first is to combine mean curvature [13] with 3D moment invariants [8] to represent the local features and global signatures of a mechanical component. The second is to change traditional region-based moment to surface-based computation to reduce the running complexity. Experiments indicate that differential moment invariants have a lower computation

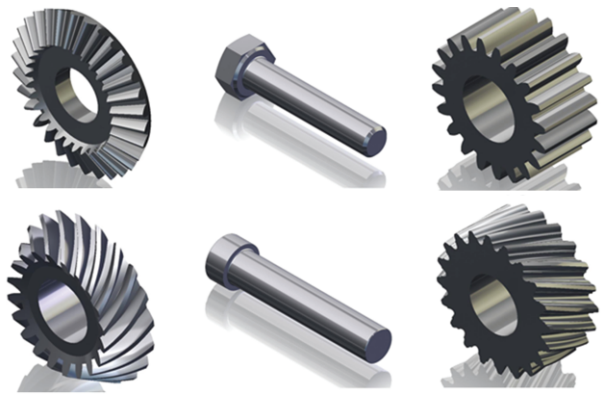


Figure 1: Similar mechanical components.

complexity than traditional 3D image retrieval approaches without reducing the retrieval rate.

2. Differential Moment Invariants

2.1. Background Information. Mean curvature can be employed for the representation of 3D mechanical component in database. In 3D Euclidean space, a parametric surface S can be represented by two fundamental forms. The two fundamental forms of a surface are uniquely determined by six parameters: E, F, G, L, M, and N [13]. Mean curvature H can be computed according to following formula:

$$H = \frac{EN + GL - 2FM}{2\left(EG - F^2\right)}. \tag{1}$$

For digital range mechanical component surfaces, approximations can be computed by local polynomial fitting approach [14].

Moment invariants have been widely used in many applications [15, 16]. To a spatial surface, 3D moment of order $p + q + r$ is defined by

$$m_{pqr} = \sum_x \sum_y \sum_z x^p y^q z^r f(x, y, z), \tag{2}$$

where (x, y, z) is the location of pixel on the surface and $f(x, y, z)$ is a function to represent the characteristic of (x, y, z). The central moment of order $p + q + r$ is defined as

$$\begin{gathered} v_{pqr} = \sum_x \sum_y \sum_z (x - x_0)^p (y - y_0)^q (z - z_0)^r f(x, y, z), \\ x_0 = \frac{m_{100}}{m_{000}}, \qquad y_0 = \frac{m_{010}}{m_{000}}, \qquad z_0 = \frac{m_{001}}{m_{000}}. \end{gathered} \tag{3}$$

The normalized moment μ_{pqr} is defined as

$$\mu_{pqr} = \frac{v_{pqr}}{v_{000}^{1-p-q-r}}. \tag{4}$$

2.2. Differential Moment Invariants. In differential moment invariants, the central moment of order $p + q + r$ will be constructed as follows:

$$v_{pqr} = \oiint_{\Sigma} (x - x_0)^p (y - y_0)^q (z - z_0)^r H(x, y, z)\, dS. \tag{5}$$

The scaling normalized moment μ_{pqr} to guarantee the scaling invariance of the central moment is introduced as follows:

$$\mu_{pqr} = \frac{v_{pqr}}{v_{000}^{p+q+r+1}}. \tag{6}$$

In order to reduce the computation cost, we choose two differential moment invariants:

$$\begin{gathered} J_1 = \mu_{200} + \mu_{020} + \mu_{002}, \\ J_2 = \mu_{200}\mu_{020} + \mu_{200}\mu_{002} + \mu_{020}\mu_{002} - \mu_{101}^2 - \mu_{110}^2 - \mu_{011}^2. \end{gathered} \tag{7}$$

2.3. Invariance Proof. This section will prove the invariance of the presented invariants. Because the mean curvature will not change under translation, so the differential moment is invariant under the translation of coordinates.

Now we consider the scaling invariance. Given a spatial discretized surface S, suppose a scaling transform with scaling factor $k > 0$:

$$\begin{aligned} v'_{pqr} &= \oiint_{\Sigma'} \left(x' - x'_0\right)^p \left(y' - y'_0\right)^q \left(z' - z'_0\right)^r \\ &\quad \times H'(x, y, z)\, dS' \\ &= \oiint_{\Sigma} (kx - kx_0)^p (ky - ky_0)^q (kz - kz_0)^r \\ &\quad \times \frac{H(x, y, z)}{k} d\left(k^2 S\right) \\ &= k^{p+q+r+1} v_{pqr}. \end{aligned} \tag{8}$$

Because $v'_{000} = \oiint_{\Sigma'} H'(x,y,z)dS' = \oiint_{\Sigma} H(x,y,z)/kdk^2S = kv_{000}$, so one can obtain:

$$\mu'_{pqr} = \frac{v'_{pqr}}{(v'_{000})^{p+q+r+1}} = \frac{k^{p+q+r+1}v_{pqr}}{(kv_{000})^{p+q+r+1}} = \mu_{pqr}. \quad (9)$$

The above formula shows μ_{pqr} is invariant to scaling transforms. So J_1 and J_2 are scaling invariances. Suppose a rotation transform is applied to the surface as follows:

$$n_i = (\cos\alpha_{i1}\cos\alpha_{i2}\cos\alpha_{i3})^T,$$
$$A = (n_1 n_2 n_3)^T, (x'y'z')^T = A(xyz)^T, \quad (10)$$

where n_i $(i = 1,2,3)$ are three orthogonal unit vectors. For simplicity, set $(x_0, y_0, z_0) = (0,0,0)$, one can obtain

$$v'_{200} = \oiint_{\Sigma'} H(x\cos\alpha_{11} + y\cos\alpha_{12} + z\cos\alpha_{13})^2 dS,$$
$$v'_{020} = \oiint_{\Sigma'} H(x\cos\alpha_{21} + y\cos\alpha_{22} + z\cos\alpha_{23})^2 dS, \quad (11)$$
$$v'_{002} = \oiint_{\Sigma'} H(x\cos\alpha_{31} + y\cos\alpha_{32} + z\cos\alpha_{33})^2 dS.$$

Under rotation transformation, the area of a surface is invariant, so $v'_{000} = v_{000}$ and $dS' = dS$. Adding the previous three formulas and considering $|A| = 1$, one can be obtain

$$\begin{aligned} J'_1 &= \frac{v'_{200} + v'_{020} + v'_{002}}{v'^{p+q+r+1}_{000}} \\ &= \frac{\oiint_{\Sigma}(x^2+y^2+z^2)H(x,y,z)\,dS}{v^{p+q+r+1}_{000}} \\ &= J_1. \end{aligned} \quad (12)$$

The invariance of J_2 can be proven through the analogy approach.

2.4. Similarity Measurement. In this paper, one-order Minkowski distance is employed to measure the difference of each two surfaces in the gallery. Let S and S' be two mechanical component curved surfaces; the one-order Minkowski distance between the two mechanical components is defined as

$$DIF(S,S') = \frac{1}{2}\sum_{i=1}^{2}\frac{|J_i - J'_i|}{\min(J_i, J'_i)}. \quad (13)$$

In addition, the average retrieval precision (ARP) [17] will be applied in the experiments to measure the precision of mechanical component retrieval.

3. Experimentation Results and Discussions

3.1. Dataset and Compare Schema. In our experiments, we have used a mechanical component dataset constructed from "http://part.3dsource.cn". In the experiment, some mechanical component categories are selected; every category contains 100 components after randomly applying translation, scaling, or rotation transformation.

Table 1: Performance of the first experiment.

Algorithm	ARP (%)	Time (ms)
Shape distribution	78.42	24635
Curvature-based method	81.75	30782
Moment invariants	70.35	10742
Fourier descriptor	72.56	28854
Neural network approach	84.24	42965
Differential moment	82.73	18536

Table 2: Performance of the second experiment.

Algorithm	ARP (%)	Time (ms)
Shape distribution	62.74	16754
Curvature-based method	65.82	18698
Moment invariants	58.68	7847
Fourier descriptor	63.58	17856
Neural network approach	73.54	27721
Differential moment	84.31	15521

In this paper, the threshold value is identified as 20% for similarity measurement. In addition, some traditional representations such as shape distribution, curvature-based method, moment invariants, Fourier descriptor, and neural network approach will be also performed for comparison. The running environment is MATLAB 7.0, 2.0 GHz CPU, and 1 GbRAM.

3.2. Common Mechanical Component Retrieval. This experimentation will test the retrieval performance for common mechanical component. The chosen 15 mechanical components categories are shown in Figure 2.

The ARP and running time of applying these approaches are shown in Table 1.

Table 1 shows that differential moment has a good retrieval precision and takes low running time. However, this method has not shown obvious advantage compared with some other approaches such as traditional moment invariants.

3.3. Similar Mechanical Component Retrieval. This experimentation will test the retrieval performance for similar mechanical component. The chosen 12 mechanical components categories are shown in Figure 3.

The ARP and running time of applying these approaches are shown in Table 2.

We can see from Table 2 that the work in this paper can get a good result. Although the traditional moment invariants take the least running time, the ARP is not good. Differential moment costs only 15521 ms in the experiment. Therefore, our approach can get better retrieval performance to mechanical components with slight different shape characteristic.

Figure 2: 15 mechanical components categories.

Figure 3: 12 mechanical components categories.

3.4. Computational Complexity Analysis. Considering a cube with N pixels edge length, the computing complexity of the algorithm can be measured as a function of N. In differential moment approach, computing the mean curvature of every pixel on surface takes $O(N^2)$ time; then differential moment invariants can be computed with $O(N^2)$; similarity measurement would only cost $O(1)$ time.

Therefore, in total, the whole running time of differential moment approach is $O(N^2)$. This is less than traditional region-based 3D moment ($O(N^3)$).

4. Conclusions

This paper combines the idea of mean curvature with traditional 3D moment invariants to solve the mechanical component retrieval problem. Experiments indicate a good result. In the future work, the research of the automatic assembly of mechanical components will be focused on.

Acknowledgments

This work is supported by Natural Science Foundation of China (61202341, 61103203), Postdoctoral Fund (2012M521552, 2012RS4054), and Postdoctoral Fund of Central South University.

References

[1] S. Wang, Y. Wang, M. Jin, X. D. Gu, and D. Samaras, "Conformal geometry and its applications on 3D shape matching,

recognition, and stitching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 7, pp. 1209–1220, 2007.

[2] S. Y. Chen and Y. F. Li, "Vision sensor planning for 3-D model acquisition," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 35, no. 5, pp. 894–904, 2005.

[3] K. Kandananond, "Using the response surface method to optimize the turning process of AISI 12L14 steel," *Advances in Mechanical Engineering*, vol. 2010, Article ID 362406, 6 pages, 2010.

[4] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: ideas, influences, and trends of the new age," *ACM Computing Surveys*, vol. 40, no. 2, article 5, 2008.

[5] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions," *ACM Transactions on Graphics*, vol. 21, no. 4, pp. 807–832, 2002.

[6] E. Ataer-Cansizoglua, E. Basb, J. Kalpathy-Cramerc, G. C. Sharpd, and D. Erdogmusa, "Contour-based shape representation using principal curves," *Pattern Recognition*, vol. 46, no. 4, pp. 1140–1150, 2013.

[7] K. H. Guo, Y. L. Liu, and G. H. Duan, "Differential and statistical approach to partial model matching," *Mathematical Problems in Engineering*, vol. 2013, Article ID 249847, 8 pages, 2013.

[8] F. A. Sadjadi and E. L. Hall, "Three dimensional moment invariants," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, no. 2, pp. 127–136, 1980.

[9] S. Y. Chen, J. Zhang, Q. Guan, and S. Liu, "Detection and amendment of shape distortions based on moment invariants for active shape models," *IET Image Processing*, vol. 5, no. 3, pp. 273–285, 2011.

[10] E. A. Lmaati, A. El Oirrak, D. Aboutajdine, M. Daoudi, and M. N. Kaddioui, "A 3-D search engine based on Fourier series," *Computer Vision and Image Understanding*, vol. 114, no. 1, pp. 1–7, 2010.

[11] B. Muller, J. Reinhardt, and M. T. Strckland, *Neural Networks: An Introduction*, Springer, Berlin, Germany, 1995.

[12] S. Chen, Z. Wang, H. Tong, S. Liu, and B. Zhang, "Optimal feature matching for 3D reconstruction by combination of global and local information," *Intelligent Automation and Soft Computing*, vol. 17, no. 7, pp. 957–968, 2011.

[13] B. Dubrovin, A. Fomenko, and S. Novikov, *Modern Geometry-Methods and Applications: Part 1*, Graduate Texts in Mathematics, Springer, Berlin, Germany, 2nd edition, 1999.

[14] P. J. Besl and R. C. Jain, "Segmentation through variable-order surface fitting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 2, pp. 167–192, 1988.

[15] C.-H. Lo and H.-S. Don, "3-D moment forms: their construction and application to object identification and positioning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 10, pp. 1053–1064, 1989.

[16] A. G. Mamistvalov, "N-dimensional moment invariants and conceptual mathematical theory of recognition n-dimensional solids," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 819–831, 1998.

[17] G.-H. Liu, Z.-Y. Li, L. Zhang, and Y. Xu, "Image retrieval based on micro-structure descriptor," *Pattern Recognition*, vol. 44, no. 9, pp. 2123–2133, 2011.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 643927, 7 pages
http://dx.doi.org/10.1155/2013/643927

Research Article

Reliability Analysis of Semiactive Magnetorheological Dampers Subjected to Harmonic Excitations

N. Mohajer Rahbari,[1] S. Talatahari,[2] and R. Aalami[3]

[1] *Structural Department, Faculty of Civil Engineering, University of Alberta, Edmonton, AB, Canada*
[2] *Marand Faculty of Engineering, University of Tabriz, Tabriz, Iran*
[3] *Department of Mechanical Engineering, University of Tabriz, Tabriz, Iran*

Correspondence should be addressed to S. Talatahari; talatahari@tabrizu.ac.ir

Received 9 April 2013; Accepted 21 May 2013

Academic Editor: Shengyong Chen

It has been seen that semiactive magnetorheological (MR) dampers are not sometimes reliable when employed in the building to contract acceleration demand of the stories. In the current paper, a three-storey sample building with semiactive MR dampers subjected to harmonic base excitation is taken under examination to clarify the reliability of such semiactive control systems in mitigation of absolute acceleration response of the buildings. Comparison between semiactively controlled building and uncontrolled original building is chosen as an implicit reliability limit-state function. Firstly, first-order reliability method (FORM) is engaged to examine the reliability of the system by linearizing the chosen nonlinear limit-state function. Afterwards, Monte Carlo simulation (MCS) is used to verify the obtained results by FORM method.

1. Introduction

Magnetorheological (MR) damper is a kind of semiactive control devices in which MR liquid is composed of magnetized tiny particles that are scattered in a mineral liquid such as silicon oil. While this liquid is exposed to a magnetic field, particles inside the liquid are polarized and chain is formed just in a few milliseconds. Hence, liquid will be changed into semi-solid state and will behave as a viscoplastic material which leads to alteration of yielding force (produced damping force).

The MR dampers are normally installed in a building so that they are resisting stories' drift. To achieve an optimal performance of the system in terms of structural control, applied voltage of the current driver must be changed to conduct the damping force close to the calculated optimal control force at any moment according to the captured response feedback being analyzed by a predefined control algorithm.

Despite promising feature of using MR dampers, it has been seen that the semiactive magnetorheological (MR) dampers are not always reliable in mitigating the acceleration demand of the stories [1]. Hence, the application of the semiactive devices to the real civil structures still remains uncommon and unreliable to the engineers. In other words, there is an open question of why the building owners should go to the expense of an additional cost for application of semiactive MR devices as they counterproductively increase the acceleration response which may lead to a massive destruction of nonstructural components.

The interesting objective of the current work is to analytically assess the reliability of semiactive MR damper control systems in reduction capability of acceleration demand for buildings subjected to base acceleration by means of existing reliability methods. For this purpose, a three-storey sample shear building is taken under examination. Modified version of Bouc-Wen model [2] is used to predict the damper's nonlinear restoring force, and clippedoptimal semiactive control algorithm [1] based on linear quadratic regulator (LQR) is updated to drive the MR damper during the control time. In terms of reliability analysis, the difference between acceleration response of controlled building and uncontrolled building resulting from time-history dynamic analysis is considered as the implicit limit-state function. This implies

Table 1: Structural properties of the case-study.

Mass of stories (m_j)	Stiffness of stories (k_j)	Periods of the first three modes (T)
$m_1 = m_2 = m_3 = 1036.8 \times 10^3$ (kg)	$k_1 = k_2 = k_3 = 456 \times 10^6$ (N/m)	0.67, 0.24, 0.17 (s)

the failure of the control system, while it cannot reduce the acceleration demand compared with uncontrolled structure. Amplitude and frequency of harmonic excitation as well as stories' stiffness are considered as random variables in reliability analysis to account for uncertainties that potentially exist in base excitation and structural natural characteristics, respectively. Well-established first-order reliability method (FORM) is applied to examine the reliability of the system by linearizing the nonlinear implicit limit-state function. Afterwards, Monte Carlo sampling (MCS) method is used to verify the obtained results by FORM analysis. All structural simulations and reliability analyses are coded in MATLAB platform.

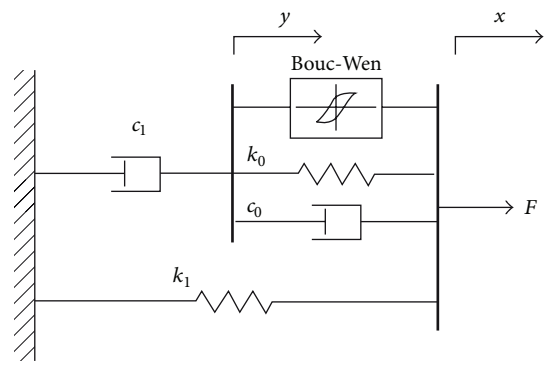


Figure 1: Mechanical model for MR damper [2].

2. Structural Model Simulation

The following sections describe the structural model and methods used to simulate the MR damper building casestudy under examination. The structural analytical model is coded in MATLAB platform.

2.1. Case-Study Introduction. An undamped three-storey shear building equipped with MR dampers is examined in this paper to evaluate the reliability of the system in acceleration response reduction. Structural properties of this building are given in Table 1.

2.2. Excitation Introduction. Because a seismic ground acceleration can be expanded by infinite numbers of harmonic excitations, in the current study, we investigate the reliability of the system in response to a harmonic base acceleration with random properties as follows:

$$\ddot{x}_g(t) = \ddot{x}_{g0} \sin(\omega t), \tag{1}$$

in which $\ddot{x}_{g0}$ and ω denote the base acceleration amplitude and frequency, respectively.

2.3. MR Damper Force. Herein, well-established Bouc-Wen phenomenological model (Figure 1) proposed by Spencer et al. [2] is utilized to predict the high nonlinear hysteretic damping force of the MR damper in the dynamic analytical model. In this case, according to Figure 1, MR damper's force is calculated from the following equation:

$$\begin{aligned} f_{\mathrm{MR}} &= \alpha z + c_0(\dot{x} - \dot{y}) + k_0(x - y) + k_1(x - x_0) \\ &= c_1 \dot{y} + k_1(x - x_0), \end{aligned} \tag{2}$$

where α is a Bouc-Wen model shape parameter related to the MR material yield stress and z is hysteretic displacement of model as given by

$$\dot{z} = -\gamma |\dot{x} - \dot{y}| z |z|^{n-1} - \beta(\dot{x} - \dot{y}) |z|^n + A(\dot{x} - \dot{y}), \tag{3}$$

in which A, n, β, and γ are the hysteretic shape parameters. According to Figure 1, $\dot{y}$ is defined by the following equation:

$$\dot{y} = \frac{1}{c_0 + c_1} \{\alpha z + c_0 \dot{x} + k_0(x - y)\}. \tag{4}$$

To determine a comprehensive model that is valid for fluctuating magnetic fields, the coefficient α, stiffness k_0 and viscous damping coefficients c_0 and c_1 are defined as a linear function of the efficient voltage u with constant coefficients as given in the following equations:

$$\begin{aligned} \alpha(u) &= \alpha_a + \alpha_b u, \qquad & k_0(u) &= k_{0a} + k_{0b} u, \\ c_0(u) &= c_{0a} + c_{0b} u, \qquad & c_1(u) &= c_{1a} + c_{1b} u. \end{aligned} \tag{5}$$

To accommodate the dynamics involved in the MR fluid reaching rheological equilibrium, the following first-order filter is employed to calculate efficient voltage u:

$$u = -\eta(u - v), \tag{6}$$

where v is the applied voltage of the current driver.

Here, one MR damper with the capacity of 1000 kN [3] is installed in such a way that causes a resistance to storey drift of the first storey. The model characteristics of the damper are reported in Table 2. Maximum input voltage ($v_{\max}$) for this damper is equal to 10 V.

2.4. System Description. The governing equation of motion of the controlled system can be expressed in state space as

$$\dot{S}(t) = A \cdot S(t) + B \cdot f(t, S) + E \cdot \ddot{x}_g(t), \tag{7}$$

in which $f(t, S)$ is the vector of dampers force and $S(t)$ denotes the state vector of the system that for an n-storey building is written as

$$\dot{S}(t) = \left\{ \frac{u(t)}{\dot{u}(t)} \right\}_{2n}, \tag{8}$$

Table 2: Model parameters for 1000 kN MR damper [3].

Parameter	Unit	Value
α_a	kN/m	46.2
α_b	kN/m/V	41.2
c_{0a}	kN·s/m	110
c_{0b}	kN·s/m/V	114.3
c_{1a}	kN·s/m	8359.2
c_{1b}	kN·s/m/V	7482.9
x_0	m	0
γ	m^{-2}	164
β	m^{-2}	164
k_0	kN/m	0.002
k_1	kN/m	0.0097
A	—	1107.2
n	—	2
η	s^{-1}	100

and A, B, and E are the constant matrices as follows:

$$A = \left[\frac{0_{n\times n}}{-M_S^{-1}K_S} \mid \frac{I_{n\times n}}{-M_S^{-1}C_S}\right]_{2n\times 2n},$$
$$B = \left[\frac{0_{n\times r}}{M_S^{-1}\Gamma}\right]_{2n\times r}, \tag{9}$$
$$E = -\left\{\frac{0_n}{\Lambda}\right\}_{2n},$$

where r demonstrates the number of stories with MR dampers, Λ is base excitation influence vector (with unit components), and Γ is dampers influence matrix with $n \times r$ dimension.

2.5. Active Controller. To achieve optimal active control force at each time instant, linear quadratic regulator (LQR) algorithm with full state feedback is designed to minimize the following cost function:

$$J = \frac{1}{2}\int_0^\infty \left((CS)^T Q\,(CS) + f^T R f\right) dt, \tag{10}$$

where Q and R are dubbed as weighting matrices whose magnitudes are defined based on the relative significance of either structural response or control forces in the optimization procedure. And C is the converter matrix.

Because the absolute acceleration of the top stories always dominant, herein, a cost function is chosen that weights the absolute acceleration of the top floor. Thus, matrix C is obtained through the following equation:

$$C = \left[-M_S^{-1}K_S \mid -M_S^{-1}C_S\right]_{n\times n}. \tag{11}$$

Numerous controllers have been tested for the three-storey example under examination, and the best result was obtained by defining the weight matrices Q and R as follows:

$$Q = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad R = 10^{-12} I_{r\times r}. \tag{12}$$

Now, the vector of optimal control force is calculated by solving the Riccati equation [4] as

$$f_{\text{opt}} = -R^{-1}B^T PS, \tag{13}$$

in which P is the Riccati matrix and S is the feedback state of the system

2.6. Semiactive Clipped-Optimal Controller. After computation of the optimal control force (f_{opt}) and calculation of the dampers force (f_{MR}) according to feedback data, to approach the MR damper control force to the desired optimal force, at each time instant, applied voltage is set as the following description.

(i) If these two forces are equal ($f_{\text{MR}} = f_{\text{opt}}$), then applied voltage is not changed.

(ii) If the absolute of MR damper force is less than the absolute of calculated optimal control force and both of them have the same sign, applied voltage should be increased to its maximum value.

(iii) Otherwise the input voltage is set to zero.

Clipped-optimal algorithm can be summarized in the following equation:

$$v = V_{\max} H\left\{\left(f_{\text{opt}} - f_{\text{MR}}\right) f_{\text{MR}}\right\}, \tag{14}$$

in which $V_{\max}$ demonstrates the maximum applied voltage that is associated with saturation of magnetic field in MR damper and $H\{\cdot\}$ is the Heaviside function.

3. Random Variables

Two sets of general uncertainties including mechanical properties of the building and base excitation properties could be mainly considered for the current problem. In terms of structural uncertainties, the major contributor is the natural frequency because the building is modeled as an idealized lumped-mass multi-degree-of-freedom (MDOF) structure. Randomness of natural frequency could result from randomness in members geometry, modulus of elasticity, or mass. However, because the contribution of higher modes could be significant due to the variation of exciting frequency, instead of taking natural frequency as the random variable, the inter-storey stiffness of stories (k_i, $i = 1, 2, 3$) considered as random variable to account for variation of natural frequency, and mass properties are held constant. Variability in structural dimensions and overall geometry tends to be small and can be adequately modeled by the normal distribution [5, 6]. However, stiffness cannot be modeled by normal distribution because it contains negative realizations. Hence, stiffness random variables are chosen to be modeled as lognormal distribution with mean of original building stiffness (Table 1) and coefficient of variation (C.O.V) of 5% because the objective of the current work is to assess the reliability of the semiactive MR damper control system for the particular problem under examination. Figure 2 indicates the lognormal probability density function (PDF) for the stiffness random variables.

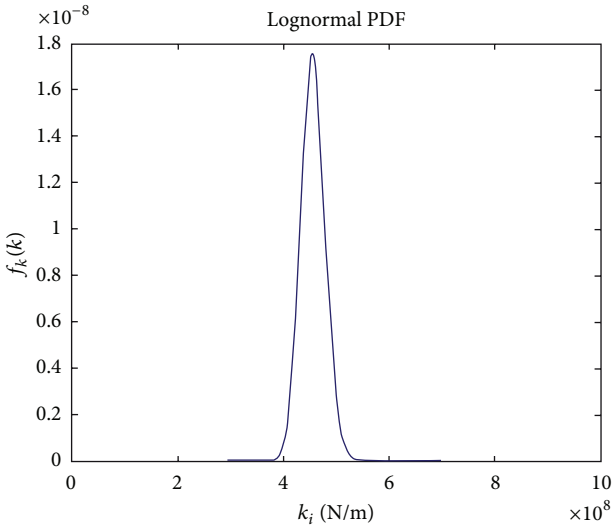


Figure 2: Lognormal distribution of stories' stiffness.

Moreover, the uncertainties for harmonic excitation come from its amplitude ($\ddot{x}_{g0}$) and frequency (ω). Since the ground peak acceleration (GPA) of an individual site is normally determined by seismic risk analysis containing lots of assumptions and uncertainties, the amplitude of the harmonic excitation is taken as a more or less wide normally distributed random variable with mean value and covariance of 0.4 g and 10%, respectively. As base excitation contains loads of harmonic contributors with a variety of frequencies, the excitation frequency is modeled as a uniform random variable to take all significant harmonic terms into consideration with the same probability of occurrence. However, because the dynamic response of the building is under examination, frequency random variables rangeing from 0.1 to 3 times first natural frequency of the building to include all major possible dynamic loads ($\omega_n + 3\sigma_{\omega_n} < \omega < \sqrt{2}\omega_n + 3\sigma_{\omega_n}$). Figure 3 illustrates the PDF for excitation amplitude and frequency. It should be verified that all five random variables are uncorrelated.

4. Limit-State Function

To study the reliability of the aforementioned semiactive system in mitigation of the absolute acceleration response of the top storey which has been chosen as the control strategy, the difference between maximum absolute acceleration of the top storey in controlled and uncontrolled modes is considered as the limit-state function. In other word, failure mode is introduced when the semiactive MR damper control system fails to reduce the acceleration demand of the top storey in comparison with the original building. Hence, the limit-state function can be expressed as

$$g(X) = \max\left(\left|\ddot{u}_{\text{uncont}}\right|\right) - \max\left(\left|\ddot{u}_{\text{cont}}\right|\right), \tag{15}$$

in which $\ddot{u}_{\text{uncont}}$ and $\ddot{u}_{\text{cont}}$ denote the maximum acceleration of the top storey for the uncontrolled and controlled builds respectively, which will be obtained through time-history analysis of both ofthe operation modes. X represents the vector of input random variables in each analysis which is defined as

$$X = \left\{\omega \;\; \ddot{x}_{g0} \;\; k_1 \;\; k_2 \;\; k_3\right\}^T. \tag{16}$$

5. Transformation of Random Variables and Jacobian Matrix

In reliability analysis, to obtain a workable space, it is usually required to repeatedly transform the realization of random variables from original distribution space into their corresponding standard normal space and vice versa. In the current problem, the random variables are independent of each other and uncorrelated. Thus, the following transformation equation is used to transform the vector of random variables X to its corresponding standard normal random variables Y [7]:

$$F_X(x) = \Phi_Y(y) \Longrightarrow \begin{cases} y = \Phi^{-1}\left[F_i(x_i)\right], \\ x = F^{-1}\left[\Phi(y_i)\right], \end{cases} \quad i = 1, 2 \ldots, 5, \tag{17}$$

where $\Phi_Y(y)$ is the cumulative distribution function (CDF) of the standard normal random variables and $F_i(x_i)$ is the original CDF of the ith random variable.

It is also necessary to obtain the transformation Jacobian to be used in transformation of gradient function for implicit limit-state function in the reliability analysis. The Jacobian matrix for uncorrelated random variables becomes a diagonal matrix obtained by [7]

$$\begin{aligned} J_{yx} &= \operatorname{diag}\left[j_{ii}\right] = \operatorname{diag}\left[\frac{\partial y_i}{\partial x_i}\right] \\ &= \operatorname{diag}\left[\frac{f_i(x_i)}{\varphi(y_i)}\right], \quad i = 1, 2 \ldots, 5, \end{aligned} \tag{18}$$

where $\varphi(y_i)$ is the PDF of the standard normal random variables and $f_i(x_i)$ is the original PDF of the ith random variable.

6. First-Order Reliability Method (FORM)

FORM reliability analysis method is based on the linearization of the limitstate at the design point which is the nearest point on the limit-state surface ($g(X) = 0$) to the origin in the standard normal space. This method is suitable for implicit limit-state functions and mostly results in conservative assessment. The FORM analysis method integrated with improved HLRF (iHLRF) algorithm [7] for finding new realization at each step of m is implemented in MATLAB platform conforming to the following steps [8].

(1) Opting a start set of random variables from the realization of their corresponding standard normal space (Y_m).

Note that for the first step, the start point can be chosen from original distribution of random variables and then transformed into standard normal space.

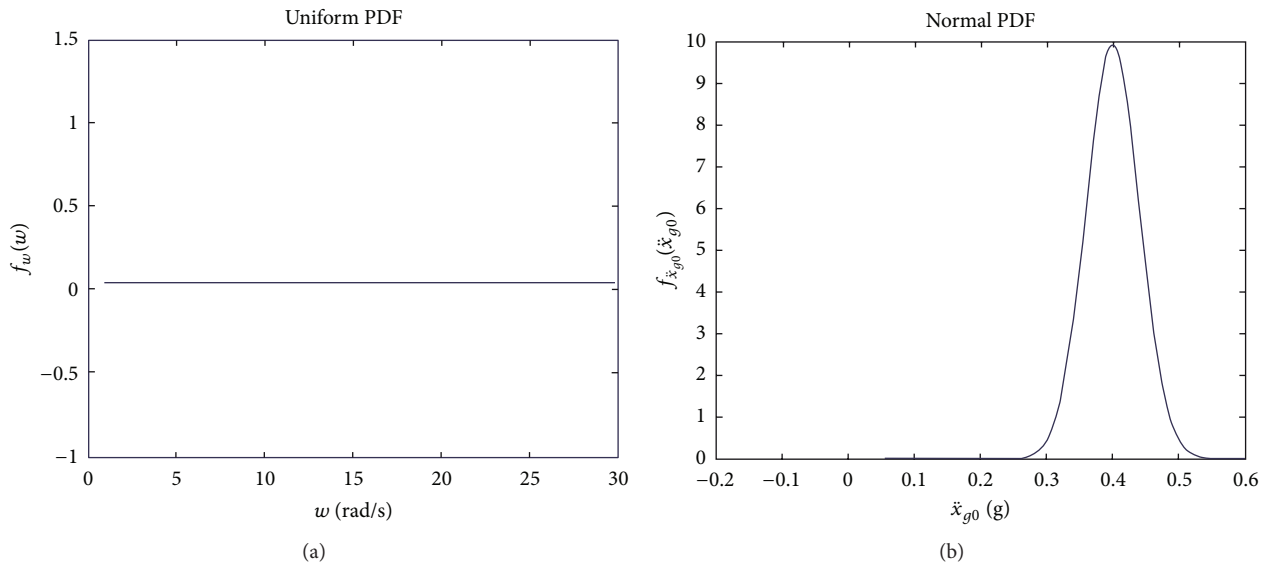


Figure 3: PDF for base excitation random variables.

(2) Transformation of Y_m into the original random variables' distribution space X_m and evaluation of limit-state function $g_m(X_m) = G_m(Y_m)$.

(3) Evaluation of the gradient (∇) of the limit-state function through the following procedure:

$$\nabla G(Y_m) = \nabla g(X_m) \equiv \frac{\partial G(Y_m)}{\partial Y} = \frac{\partial g(X_m)}{\partial Y}, \quad (19)$$

where for the current implicit function which is a function of maximum acceleration responses it comes to

$$\frac{\partial g(X)}{\partial Y} = \nabla g_{\ddot{U}}{}^T \times \nabla \ddot{U}_X \times J_{yx}^{-1} = \frac{\partial g^T}{\partial \ddot{U}} \times \frac{\partial \ddot{U}}{\partial X} \times J_{yx}^{-1}, \quad (20)$$

in which $\ddot{U}$ is the response vector defined as

$$\ddot{U} = \{\max(|\ddot{u}_{\text{uncont}}|) \ \ \max(|\ddot{u}_{\text{cont}}|)\}^T, \quad (21)$$

and gradient of response with respect to the random variables ($\nabla \ddot{U}_X = \partial \ddot{U}/\partial X$) is numerically obtained by taking finite differences of $\Delta x_i = 0.001\sigma_{x_i}$ at each step of analysis, where σ_{x_i} denotes the standard deviation of random variable x_i.

(4) Examination of the following convergence criteria to verify that the found design point is being located on the limit-state surface while it possesses the closest distance from origin:

$$\left|\frac{G_m(Y_m)}{G_1(Y_1)}\right| \le 10^{-3}, \qquad \left\|Y_m - \left(\alpha_m^T Y_m\right)\alpha_m\right\| \le 10^{-3}, \quad (22)$$

in which

$$\alpha_m = -\frac{\nabla G(Y_m)}{\|\nabla G(Y_m)\|}. \quad (23)$$

(5) If convergence is satisfied, then probability index (β) and probability of failure (P_f) are obtained as

$$\beta = \|Y_m^*\|, \qquad P_f = \Phi(-\beta). \quad (24)$$

If convergence is not met, the new random variables in standard normal space are found as

$$Y_{m+1} = Y_m + s_m d_m, \quad (25)$$

in which s_m is the step size chosen as unity and d_m is the search direction defined by

$$d_m = \left(\frac{G_m(Y_m)}{\|\nabla G(Y_m)\|} + \alpha_m^T Y_m\right)\alpha - Y_m, \quad (26)$$

and procedure is iteratively repeated from step 2 until the convergence criteria are satisfied.

7. Monte Carlo Simulation (MCS)

MCS reliability approach is based on repeated sampling from the linearization of the random variables and examining whether the limit-state function is located in failure region or not. Then, probability of failure is calculated based on taking these samples as independent random variables with equal probability of occurrence. The MCS method is coded in MATLAB platform following the following steps [8].

(1) A set of random standard normal variables (Y_k) is generated according to joint standard normal PDF as

$$\varphi(Y) = \frac{\exp\left(-0.5Y^T Y\right)}{(2\pi)^{0.5N}}, \quad (27)$$

in which N denotes the number of random variables.

Table 3: Results of FORM analysis.

Start point	$M_x - 3\sigma$	$M_x - 2\sigma$	$M_x - 1.5\sigma$	$M_x - \sigma$	M_x	$M_x + \sigma$	$M_x + 1.5\sigma$	$M_x + 2\sigma$	$M_x + 3\sigma$
Iteration	14	11	10	15	8	11	*23*	9	*45*
e_1	$2.26E-04$	$3.26E-06$	$3.11E-04$	$9.16E-06$	$5.99E-07$	$1.55E-04$	*$8.17E-08$*	$5.70E-05$	*$8.44E-07$*
e_2	$4.34E-04$	$1.69E-05$	$3.73E-04$	$4.30E-05$	$1.93E-05$	$2.25E-04$	*$3.41E-04$*	$7.84E-05$	*$4.66E-04$*
β	0.990	0.807	0.949	0.990	0.807	0.990	*1.310*	0.990	*2.121*
P_f (%)	**16.10**	**20.99**	**17.12**	**16.10**	**20.99**	**16.10**	*9.52*	**16.10**	*1.70*
	−0.9899	−0.8066	−0.9491	−0.9899	−0.8066	−0.9899	*1.2046*	−0.9899	*−2.0843*
	0.0045	0.0032	0.0009	0.0041	0.0032	0.0043	*0.1751*	0.0041	*−0.0824*
Y	−0.0202	−0.0060	−0.0182	−0.0201	−0.0060	−0.0200	*0.2650*	−0.0201	*−0.0536*
	0.0008	−0.0006	−0.0103	0.0007	−0.0006	0.0008	*0.2297*	0.0007	*−0.0443*
	0.0112	0.0089	0.0020	0.0111	0.0089	0.0112	*0.3321*	0.0111	*0.3766*

Table 4: Random variables' values in their original space corresponding to design point.

ω	$\ddot{x}_{g0}$	k_1	k_2	k_3
5.596678	3.92561	$4.55E+08$	$4.55E+08$	$4.56E+08$

(2) Generated random standard normal numbers (Y_k) are transformed to their original distribution space (X_k).

(3) Limit-state function is evaluated $g_k(X_k)$. And indicator function is defined as

$$I_k(X_k) = 1 \quad \text{if } g_k(X_k) < 0, \qquad I_k(X_k) = 0 \quad \text{if } g_k(X_k) \geq 0 \tag{28}$$

(4) The convergence criterion for covariance of failure probability (δ_{P_f}) is checked as follows:

$$\delta_{P_f} = \frac{1}{\sqrt{K}} \cdot \frac{\sigma_I}{\mu_I} \leq 0.02 \tag{29}$$

in which K is the total samples number and σ_I and μ_I represent standard deviation and mean values for indicator vector $I = \{I_k(X_k)\}_K$, $k = 1, 2, 3 \ldots, K$, respectively.

(5) If the convergence criterion is achieved, the probability of failure is calculated as

$$P_f = \mu_I. \tag{30}$$

If not, more samples are required and the procedure is iteratively followed from step 1.

8. Numerical Results

8.1. FORM. Table 3 demonstrates the results obtained from FORM analysis. Due to inherent nonlinearity existing in limit-state function, various start points including mean values and mean values (M_x) plus or minus 1 to 3 standard deviations (σ) were engaged to verify the robustness of the design point over the whole domain. As it can be seen, the FORM analysis converges to local design points for start points $M_x + 1.5\sigma$ and $M_x + 3\sigma$. However, the rest of the 7 start points converge to nearly the same point which is the global design point for the introduced limit-state function. Hence, the dominant probability of failure obtained by FORM analysis comes to 16.10%. Table 4 depicts the design point values for random variables in their original space. It is apparently seen that the stiffness of the stories seems to possess less importance in reliability analysis, but the excitation frequency has the most importance.

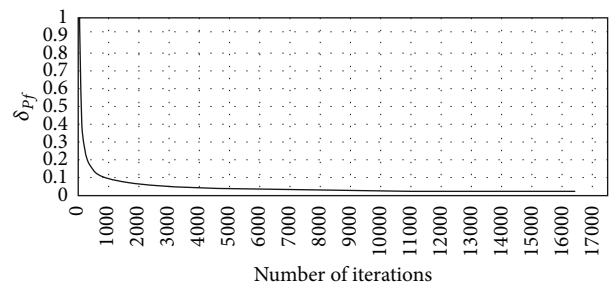


Figure 4: Covariance growth of failure probability.

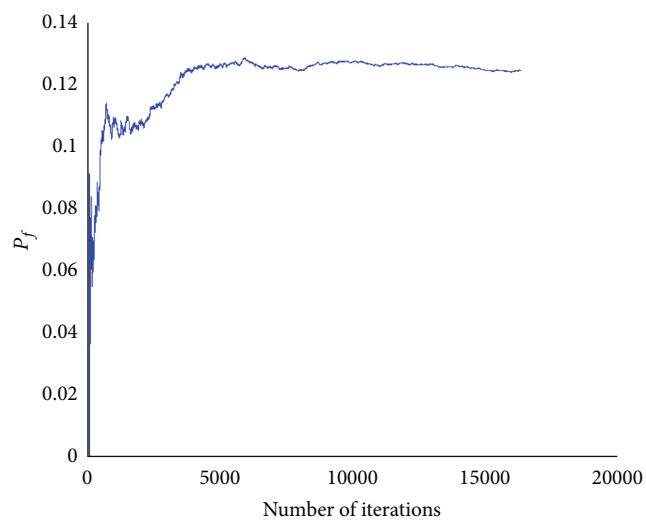


Figure 5: Failure probability changes over iterations.

8.2. MCS. Figure 4 illustrates the covariance of failure probability (δ_{P_f}) against number of iterations, and Figure 5 shows

the changes of failure probability against iterations resulting from Monte Carlo simulation. The analysis converged at 17475th iteration with 1.9996% coefficient of variation for failure probability, and the probability of failure obtained equals 12.52%.

9. Conclusions

Reliability of semiactive MR damper control system in mitigation of absolute acceleration of a three-storey building was studied in the current work. From the analyses conducted by FORM and MCS methods, it can be concluded that MR dampers are almost reliable by more than 83.9% in terms of acceleration reduction of building due to base acceleration. The result achieved by MCS agrees with the correctness of the outcome of FORM analysis. MSC gives more accurate result by taking far more computation efforts compared with FORM analysis. However, FORM analysis develops a good conservative result just after finite number of iterations which means that FORM analysis could be effectively employed for this kind of problems.

Similar limit-state functions to (15) can also be defined for the other structural responses (e.g., interstorey drift and displacement responses) to obtain a more comprehensive reliability assessment. In this case, obtained results from FORM or MCS analyses for each limit-state function are considered as a "component reliability analysis." Afterwards, component reliability analyses are integrated by means of "system reliability analysis" methods to acquire the whole system reliability [7]. Hence, system reliability analysis of the MR damper building is suggested as a future work.

References

[1] S. J. Dyke, B. F. Spencer Jr., M. K. Sain, and J. D. Carlson, "Modeling and control of magnetorheological dampers for seismic response reduction," *Smart Materials and Structures*, vol. 5, no. 5, pp. 565–575, 1996.

[2] B. F. Spencer Jr., S. J. Dyke, M. K. Sain, and J. D. Carlson, "Phenomenological model for magnetorheological dampers," *Journal of Engineering Mechanics*, vol. 123, no. 3, pp. 230–238, 1997.

[3] H.-J. Jung, B. F. Spencer Jr., and I.-W. Lee, "Control of seismically excited cable-stayed bridge employing magnetorheological fluid dampers," *Journal of Structural Engineering*, vol. 129, no. 7, pp. 873–883, 2003.

[4] T. T. Soong, *Active Structural Control: Theory and Practice*, John Wiley & Sons Inc, New York, NY, USA, 1990.

[5] A. Moustafa and S. Mahadevan, "Reliability analysis of uncertain structures using earthquake response spectra," *Earthquake and Structures*, vol. 2, no. 3, pp. 279–295, 2011.

[6] Joint Committee on Structural Safety (JCSS), *Probabilistic Model Code*, Technical University of Denmark, Denmark, 2013.

[7] A. D. Kiureghian, "Chapter 14: First- and second-order reliability methods," in *Engineering Design Reliability Handbook*, E. Nikolaidis, D. M. Ghiocel, and S. Singhal, Eds., CRC Press, Boca Raton, Fla, USA, 2005.

[8] T. Haukaas and A. D. Kiureghian, "Finite Element Reliability and Sensitivity Methods for Performance-Based Earthquake Engineering," Pacific Earthquake Engineering Research Center, College of Engineering, University of California, Berkeley, Calif, USA, 2004, http://peer.berkeley.edu/publications/peer_reports/reports_2003/0314.pdf.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 392531, 15 pages
http://dx.doi.org/10.1155/2013/392531

Research Article

Comparison of Soft Computing Techniques for Modelling of the EDM Performance Parameters

M. V. Cakir,[1] O. Eyercioglu,[2] K. Gov,[2] M. Sahin,[3] and S. H. Cakir[3]

[1] *Mechanical Department, Vocational School of Kilis, Kilis 7 Aralik University, Kilis 79000, Turkey*
[2] *Department of Mechanical Engineering, Gaziantep University, Gaziantep 27310, Turkey*
[3] *Department of Mathematics, Gaziantep University, Gaziantep 27310, Turkey*

Correspondence should be addressed to M. V. Cakir; cakir@kilis.edu.tr

Received 13 March 2013; Accepted 13 May 2013

Academic Editor: Amir Hossein Alavi

Selection of appropriate operating conditions is an important attribute to pay attention for in electrical discharge machining (EDM) of steel parts. The achievement of EDM process is affected by many input parameters; therefore, the computational relations between the output responses and controllable input parameters must be known. However, the proper selection of these parameters is a complex task and it is generally made with the help of sophisticated numerical models. This study investigates the capacity of Adaptive Nero-Fuzzy Inference System (ANFIS), genetic expression programming (GEP) and artificial neural networks (ANN) in the prediction of EDM performance parameters. The datasets used in modelling study were taken from experimental study. According to the results of estimating the parameters of all models in the comparison in terms of statistical performance is sufficient, but observed that ANFIS model is slightly better than the other models.

1. Introduction

Recently, as a result of advances in technology, the use of a variety of materials with superior properties has become commonplace. Manufacturing of these materials is extremely difficult or impossible via traditional machining process; however, EDM is a mostly used process for manufacturing such parts [1]. In EDM process, material is removed rapidly and repeatedly by spark discharges across the gap between the workpiece and the tool, both of them are immersed in a dielectric. Surface roughness (R_a), material removal rate (MRR), and electrode wear ratio (EWR) are the most important output parameters from the point of efficiency and quality. Many input parameters affect these performance parameters; therefore, the appropriate selection of the input parameters is needed.

Determining the optimum process parameters is important in terms of increasing the surface quality, reducing the processing time, and the tools wear [2]. But metal removal process in EDM has a nonlinear, randomness, and time varying properties [3]. EDM process is requiring much skill or effort but it has a quite complex mechanism and its mechanism is still not exactly understood. The application area of EDM process is limited due to its disadvantages such as lower productivity, a high specific energy consumption, and longer processing time. To create a model that can predict EDM performance by using input parameters with high accuracy is very difficult. However, a quantitative relationship between the performance parameters and controllable input parameters is often needed in EDM. Mathematical models help to know the effects of different process variables on the output performances and enhance the operating performance by applying some of the theoretical outcomes [4]. Therefore, researches are concentrating on the optimization and the process modeling of EDM in order to develop the productivity and finishing capability in last decades. Some regression methods have been applied for modelling the EDM process [5]. Because of the advantages of the artificial intelligence systems, many researchers studied to find

Table 1: Machining conditions in EDM.

	Levels
Variable parameters	
Discharge current, I (A)	3, 6, 12.5, 25, 50
Pulse on-time, T_{on} (μs)	3, 6, 12, 24, 48, 99, 201, 402, 801, 1602
Pulse off-time, T_{off} (μs)	3, 6, 12, 24, 48, 99, 201
Constant parameters	
Dielectric	Kerosene
Gap voltage (volt)	60–80 V
Dielectric flushing pressure	0.1 MPa
Polarity of tool electrode	Positive, +

the relationships between input and output parameters in EDM process by using soft computing techniques.

Indurkhya and Rajurkar [6] presented an ANN model. MRR and surface roughness estimating capability of different ANN algorithms were investigated by Tsai and Wang [7, 8]. Lin et al. investigated the grey relational analysis based on an orthogonal array and fuzzy-based Taguchi method for optimizing the EDM process [9]. A hybrid ANN and GA methodology to modeling and optimization of EDM were used by Wang et al. [10]. An artificial feed forward NN based on the Levenberg-Marquardt backpropagation technique was developed to estimate the MRR by Panda and Bhoi [11]. For the selection of EDM process parameters, Yilmaz et al. developed a fuzzy logic-based system [12]. Mandal et al. used an ANN and backpropagation algorithm for modelling and optimizing the EDM process [2]. Salman and Kayacan developed a model between EDM inputs and R_a by using GEP [13]. Rao et al. developed a hybrid model using ANN and GA for modelling and optimization of surface roughness (R_a) in EDM [14]. Maji and Pratihar used (ANFIS) to establish input-output relationships of an electrical discharge machining both in forward as well as reverse directions in 2010 [15]. Joshi and Pande reported an intelligent approach for process modelling and optimization of (EDM) [16]. Though many modelling studies for EDM process have been done until now, these models are based on several assumptions and they cannot be generalized to all conditions; therefore, modelling studies will go on in the future.

However, a comprehensive study to compare the performances of GEP, ANN, and ANFIS for modelling of important EDM performances is still missing. The aim of this study is to compare the mostly used soft computing techniques which are ANN, ANFIS, and GEP in prediction accuracy of the EDM process. In this study, soft computing techniques are applied on the outcomes of experimental works. Discharge current, pulse on-time, and pulse off-time were taken as input parameters and surface roughness, electrode wear ratio, and material removal rate are taken as the output parameters. Using the experimental dataset, ANFIS, ANN, and GEP models are developed and then their performances were compared.

2. Experimental Studies

The die sinking EDM machine (Furkan Model M50A) with servo head (constant gap) is used to conduct the experiments. Twelve mm depth blind holes were formed using different parameters. Electrode was copper with 20 mm diameters and dielectric liquid was kerosene. During the experiments, pulse off-time (T_{off}), pulse on-time (T_{on}), and discharge currents (I) were taken as variable parameters. Gap voltage, flushing pressure, and polarity were kept constant. Values of these parameters are shown in Table 1. The work material used in this study was DIN 1.2080 tool steel. The dimension of the work material was 60 mm in diameter and 30 mm in height.

Machining performances of EDM are influenced be various process parameters and these parameters can be varied in each stage of EDM. Selection of variable and constant input parameters is based on work experience, handbook values, and researchers' reports [17–19]. General EDM performances parameters are surface roughness, MRR, and EWR except for the electrode material type and dielectric fluid [17].

MRR in EDM is an important factor to estimate the time of finishing the machined part. MRR in EDM process is an important factor because of its vital effect on the industrial economy. MRR is commonly described as material volume per min (mm^3/min) and calculated as the removed mass from the workpiece material during the machining by dividing into the machining time and density of workpiece. MRR value is calculated from the following equation:

$$\text{MRR} = \frac{W_f - W_i}{t \times \rho} = \text{mm}^3/\text{min}, \tag{1}$$

where W_i = initial mass of workpiece material (g), W_f = final mass of workpiece material (g), t = operating time (min), and ρ = density of workpiece (g/mm^3). Accuracy of the product size is influenced by electrode wear and therefore much research is being done to minimize electrode wear. Minimum discharge current gives minimum electrode wear but in that case MRR will decrease. Conversely the high current gives a high MRR with high electrode wear. Completely elimination of electrode wear is not possible, but it can be reduced by selecting optimum settings of input parameters. The ratio of loss weight of the electrode to the loss weight of workpiece material is defined as EWR and mostly expressed as a percentage as

$$\text{EWR}\,(\%) = \frac{\text{Loss weight of electrode}}{\text{Loss weight of workpiece}} \times 100. \tag{2}$$

One of the most important quality parameters of the EDM process is workpiece surface roughness. Average surface roughness (R_a) among the various amplitude parameters is mostly used in EDM process. In this study, in order to measure surface roughness values of machined samples, Mahr stylus instrument (MarSurf XR 20 with GD 25) available

Table 2: Input and output parameters of GEP models.

	Code	Variables	Symbol	Range
Input	d0	Discharge current (Ampere)	I	3–50
Input	d1	Pulse on-time (μs)	T_{on}	3–1602
Input	d2	Pulse off-time (μs)	T_{off}	3–201
Output 1	F1	Material removal rate (mm^3/min)	MRR	3.4–390
Output 2	F2	Electrode wear ratio (%)	EWR	0.01–45
Output 3	F3	Surface roughness (μm)	R_a	1.4–18

Table 3: GEP model parameters.

P1	Function set	ln, log, $1/X$, tan, $\sqrt[3]{}$, X^3, X^2, e^x, $\sqrt{}$, /, $*$, −, +,
P2	Gene number	5, 4, 3, 2, 1
P3	Head size	15, 10, 8, 5
P4	Linking function	(+), ($*$)
P5	Number of generation	10000–20000
P6	Chromosomes	28–45
P7	Mutation rate	0.044
P8	Inversion rate	0.1
P9	One-point recombination rate	0.3
P10	Two-point recombination Rate	0.1
P11	Gene recombination rate	0.1
P12	Gene transposition rate	0.1

was used. Experimental results of three performances and prediction of all models are given at Tables 13–17.

3. Modelling Study

The artificial intelligence approaches have lately emerged as promising for developing a useful model for the complex systems. The purpose of this study is to present a proficient and incorporated approach to establish a model that can accurately predict the performances of EDM process by correlating the input parameters. Therefore, the well-known soft computing methods such as ANFIS, GEP, and ANN were compared for predicting performances. Thus the precision, efficiency, and quality of EDM process can be improved by choosing the best model.

3.1. Development of Genetic Programming (GP) Models. GP is an artificial intelligence method which has many application areas like engineering, science, and industrial and mechanical models [20–22]. GP proffers a solution as a result of the evolution of computer programs by using natural selection methods. Genetic programming (GP) was developed by Koza [23]. GP is a propagation of the traditional genetic algorithm (GA). Ferreira invented an evolutionary algorithm called gene expression programming (GEP) to predict mathematical models by using experimental data in 2001 [24]. GEP includes simple linear chromosomes of fixed length like that used in GA and forked forms of various sizes and shapes similar to the parse trees of GP [25].

Table 4: Function set's list.

Code	Function Set
S1	tan, $\sqrt[3]{}$, X^3, X^2, log, ln, e^x, $\sqrt{}$, /, $*$, −, +
S2	$\sqrt[3]{}$, X^3, X^2, log, ln, e^x, $\sqrt{}$, /, $*$, −, +
S3	log, ln, e^x, $\sqrt{}$, /, $*$, −, +
S4	e^x, $\sqrt{}$, /, $*$, −, +
S5	/, $*$, −, +

In this study GeneXproTools 4.0 program was used to develop mathematical models which are used to estimate MRR, EWR, and R_a values. Primarily the chromosomes of a number of individuals are produced fortuitously in the GEP process. Then the chromosomes are stated as expression trees (ETs), evaluated based on a fitness function, and selected by fitness to reproduce with modification by means of genetic operations. The new generation of solutions experiences the same process and the process is repeated until the stop condition is fulfilled. The fittest individual is used as the final solution [26].

Experimental parameters and their ranges are shown in Table 2, and parameters which are used for construction of GEP models are specified in Table 3. The list of function set is presented in Table 4. The data obtained from experimental study is separated into two parts as training and test sets to find out explicit formulations. The results of GEP models obtained from different combinations are presented in Table 5. Longer computational time was needed for training all of these combinations. Hence a subset of these combinations was chosen to analyse the GEP performance's algorithm for estimating the MRR, EWR, and R_a. The optimal settings obtained from GEP models are shown as bold in Table 5. Consequently, these optimal settings are used for the estimation of all performances.

In this study three statistical parameters were used for measuring performance of GEP models. They are coefficient of correlation (R^2), mean square error (MSE), and mean absolute error (MAE). Table 6 shows the best statistical results obtained from GEP formulation.

Test and training datasets used for construction of all models are selected randomly. Regarding the MRR, EWR, and R_a formulation, 30 training data and 10 test data were used. The offered GP formulations must be suitable range as remarked in Table 2. The formulations resulting from training values are checked by test values. The GEP modelling

Table 5: The results obtained from GEP models.

					MRR R^2 error			EWR R^2 error			R_a R^2 error	
P1	P2	P3	P4	P5	Train	Test	P5	Train	Test	P5	Train	Test
1	2	5	+	12115	0.992	0.986	12169	0.980	0.983	12964	0.966	0.964
S1	2	5	*	18751	0.994	0.979	19907	0.915	0.923	10272	0.993	0.990
S1	3	8	+	18725	0.995	0.987	15916	0.947	0.923	11373	0.993	0.992
S1	3	8	*	17185	0.987	0.961	12170	0.967	0.978	11268	0.968	0.976
S1	4	10	+	18022	0.995	0.995	15130	0.918	0.900	11514	0.977	0.957
S1	4	10	*	14363	0.993	0.990	13788	0.919	0.820	13170	0.992	0.988
S1	5	15	+	14628	0.992	0.992	15273	0.941	0.910	19517	0.990	0.983
S1	5	15	*	18125	0.995	0.988	14868	0.869	0.871	17871	0.991	0.993
S2	2	5	+	17574	0.993	0.989	12716	0.954	0.964	12239	0.992	0.993
S2	2	5	*	18880	0.991	0.990	17180	0.900	0.891	11899	0.992	0.991
S2	3	8	+	17606	0.993	0.991	11404	0.980	0.862	10227	0.984	0.985
S2	3	8	*	18607	0.991	0.889	15687	0.981	0.932	10314	0.991	0.991
S2	4	10	+	13058	0.988	0.987	12108	0.941	0.907	14376	0.989	0.979
S2	4	10	*	18954	0.957	0.955	12564	0.854	0.868	11353	0.984	0.982
S2	5	15	+	14440	0.994	0.991	18056	0.941	0.923	19195	0.991	0.987
S2	5	15	*	12960	0.993	0.979	12577	0.926	0.904	17192	0.903	0.948
S3	2	5	+	14480	0.991	0.985	13080	0.952	0.957	10129	0.987	0.988
S3	2	5	*	14780	0.994	0.994	14465	0.956	0.916	11534	0.992	0.992
S3	3	8	+	14169	0.989	0.985	16631	0.981	0.891	10807	0.992	0.990
S3	3	8	*	13686	0.995	0.992	14565	0.974	0.927	11011	0.971	0.955
S3	4	10	+	17130	0.995	0.989	11764	0.944	0.910	10863	0.983	0.981
S3	4	10	*	18529	0.994	0.864	11405	0.961	0.938	13042	0.991	0.992
S3	5	15	+	11502	0.994	0.994	17243	0.926	0.905	18962	0.981	0.949
S3	5	15	*	11890	0.986	0.975	17553	0.927	0.913	19290	0.993	0.993
S4	2	5	+	12340	0.981	0.981	11981	0.977	0.965	15561	0.991	0.990
S4	2	5	*	13152	0.993	0.992	10536	0.978	0.986	14886	0.985	0.764
S4	3	8	+	12879	0.985	0.982	15429	0.955	0.947	13396	0.973	0.968
S4	3	8	*	15082	0.995	0.978	11563	0.981	0.980	13043	0.992	0.994
S4	***4***	***10***	+	18092	0.990	0.987	10491	0.909	0.908	***10552***	***0.993***	***0.994***
S4	4	10	*	19587	0.993	0.992	18350	0.885	0.820	13543	0.982	0.989
S4	5	15	+	12021	0.995	0.993	12543	0.928	0.930	17273	0.991	0.977
S4	5	15	*	12165	0.972	0.992	16252	0.680	0.780	16555	0.990	0.888
S5	***2***	***5***	+	***19635***	***0.996***	***0.995***	18978	0.976	0.976	15273	0.979	0.980
S5	2	5	*	12069	0.993	0.994	11267	0.977	0.979	15865	0.900	0.896
S5	3	8	+	14075	0.995	0.989	12766	0.809	0.765	11841	0.815	0.898
S5	***3***	***8***	*	18934	0.995	0.993	***16346***	***0.983***	***0.988***	12751	0.951	0.903
S5	4	10	+	17575	0.995	0.987	10901	0.975	0.937	10802	0.987	0.820
S5	4	10	*	15170	0.990	0.995	16341	0.903	0.768	13875	0.948	0.945
S5	5	15	+	12506	0.992	0.965	19621	0.956	0.914	12004	0.973	0.950
S5	4	10	*	11042	0.975	0.963	10924	0.821	0.801	17834	0.977	0.968

Table 6: Best statistical results obtained from GEP formulation.

Statistical parameters	(MRR)		(EWR)		(R_a)	
	Training	Testing	Training	Testing	Training	Testing
MSE	55.351	162.23	3.5504	3.6266	0.0966	0.2201
MAE	5.4858	8.3515	1.2967	1.2536	0.2361	0.3378
R^2	0.996	0.995	0.983	0.988	0.993	0.994

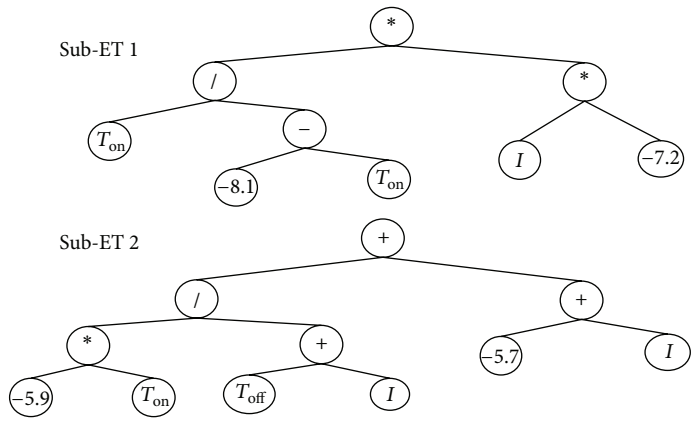


Figure 1: Expression tree for MRR.

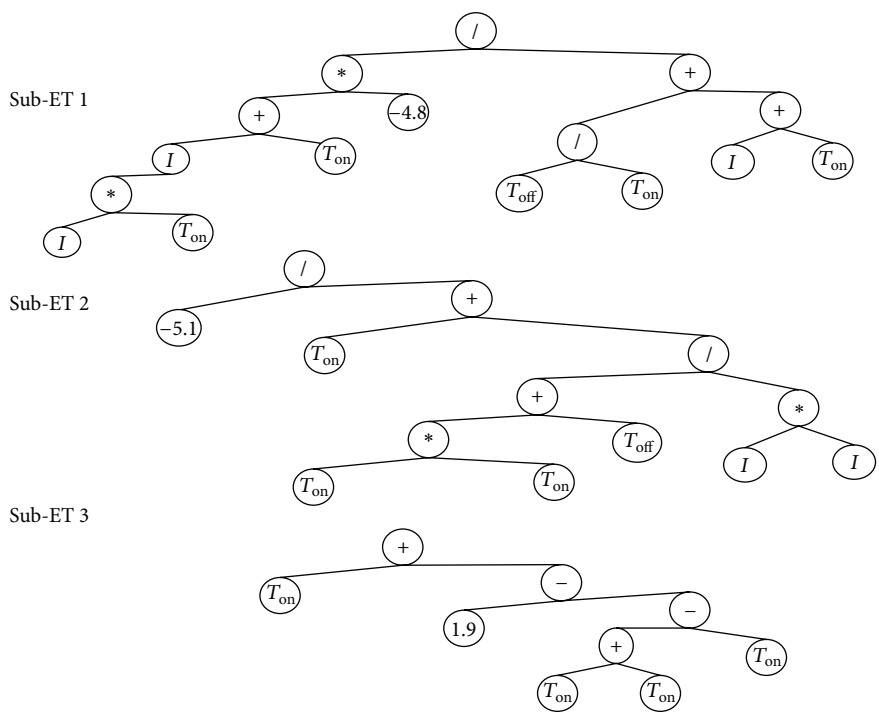


Figure 2: Expression tree for EWR.

results in the expression tree form are presented in Figures 1, 2, and 3 which correspond to the following equations:

$$\mathrm{MRR} = \frac{7.2 \times I \times T_{\mathrm{on}}}{8.1 + T_{\mathrm{on}}} + \frac{-5.9 \times T_{\mathrm{on}}}{T_{\mathrm{off}} + I} + I - 5.7,$$

$$\mathrm{EWR} = \frac{-4.8 \times \left(I \times T_{\mathrm{on}} + T_{\mathrm{on}}\right)}{\left(\left(T_{\mathrm{off}}/T_{\mathrm{on}}\right) + I + T_{\mathrm{on}}\right)} \times \frac{-5.1}{T_{\mathrm{on}} + \left(\left(T_{\mathrm{on}}^2 + T_{\mathrm{off}}\right)/I^2\right)} \times 1.93,$$

$$\begin{aligned} R_a = {} & \left(2 \times T_{\mathrm{on}} \times I\right)^{1/4} - T_{\mathrm{on}} - 2 \times T_{\mathrm{off}} \\ & + T_{\mathrm{off}} - e^{\left[-8.7 - e^{I - T_{\mathrm{off}}} - 8.6\right]} \\ & + T_{\mathrm{off}} - e^{\left[I - e^{I+1.9} + 0.1 + I\right]} \\ & + T_{\mathrm{on}}\ e^{\left[I - \left(\left(T_{\mathrm{on}} - I\right)/\left(I - 9\right)\right) + I + T_{\mathrm{on}}\right]}. \end{aligned} \tag{3}$$

3.2. Development of Artificial Neural Networks (ANN) Models. An artificial neural network is a model which runs like

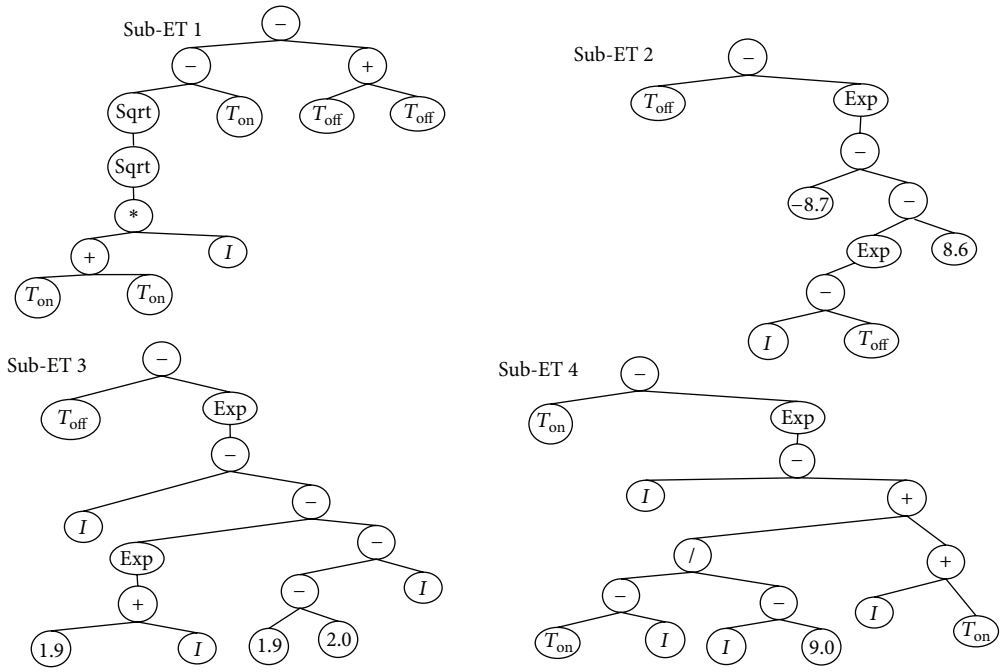


Figure 3: Expression tree for R_a.

Table 7: ANN model's specifications.

Paradigm	Supervised
Learning rule	Error-correction (Levenberg-Marquardt)
Architecture	Feed-forward multilayer perceptron
Learning algorithm	Backpropagation
Performance function	MSE
Model	3-4-1 and 3-5-1
Neurons of input layer	3
Number of hidden layer	1
Neurons of hidden layer	4 and 5 (compared to each)
Activation of hidden layer	TanhAxon
Neurons of output	1
Activation of output layer	TanhAxon

a human brain by using many neurons consecutively. The network collects information by a learning process [26]. Complex problems whose analytical or numerical solutions are difficult to be obtained and can be solved by utilizing adaptive learning ability of neural networks [27]. Neural networks are generally classified according to their network topology (i.e., feedback or feedforward) and learning or training algorithms (i.e., supervised or unsupervised). Multilayer perceptron (MLP) which is the most common neural network model is a feedforward neural network with one or more hidden layers [28]. MLP is a supervised type network as it needs a desired output to learn. The main advantage of MLP is that it is easy to use and its capability of approximating arbitrary functions in terms of mapping abilities. The requirement of longer training time and lots of training data (typically three times more training samples than network weights) are the main disadvantages.

Normally, the network consists of an input layer of source neurons, at least one middle layer or hidden layer of computational neurons, and an output layer of computational neurons. A multilayer perceptron with one hidden layers is shown in Figure 4. Each layer in a multilayer neural network has its own specific function. The function of input layer is accepting the input signals from environment and redistributing these signals to each neuron in the hidden layer. The function of output layer is receiving the output signals from the hidden layer and creating the output model of the entire network. Neurons in the hidden layer perceive the features; the weights of the neurons represent the features hidden in the input patterns. These features are then used for determining the output pattern by the output layer.

In this study, an ANN model was designed in MATLAB (version 7.0) environment. This developed ANN model's specifications are listed in Table 7. The developed ANN model

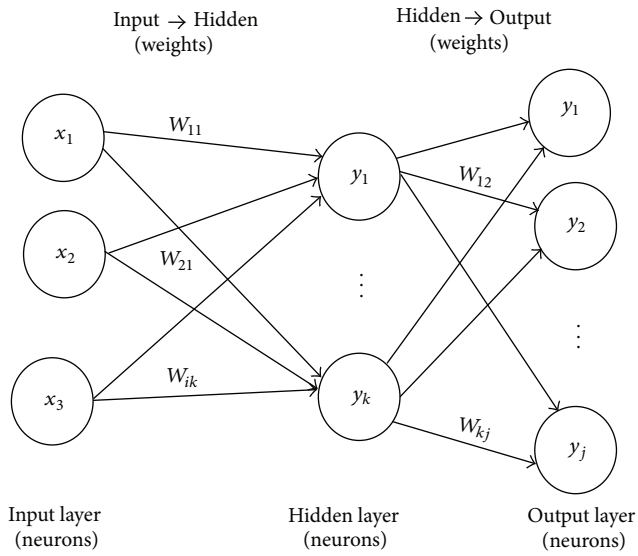


Figure 4: Three-layered feed forward neural network topology.

consists of one input layer, one hidden layer, and one output layer. The inputs to be used in the ANN are the same as the ones in GEP models.

Firstly training and test data sets were created, and then the best structure of MLP for this application was developed by training of different types of transfer and output functions for layers and also different numbers of hidden layers and its neurons were developed as well, and at the end, the neural network model was established by comparing the network prediction to validation data (10% of the data). In this study, the neuron numbers in the hidden layer were used as a main parameter to obtain the best ANN structure. Therefore, they were changed until achieving the minimum mean squared error. Regarding the MRR, EWR, and Ra formulation, 30 training, 4 validation, and 6 tests data were used as training and test sets. From this trial and error study, TanhAxon function was found tobe the most compatible one for the hidden layer and output layer in all tested ANN models. TanhAxon squashes the range of each neuron in the layer to between −1 and 1. Such nonlinear elements provide a network with the ability to make soft decisions. The output of the jth neuron after activation can be evaluated by using following equation:

$$f(x) = \frac{e^{2x} - 1}{e^{2x} + 1}, \quad f(x) \in (-1, 1). \tag{5}$$

Before training ANN, input data set was scaled between 0 and 1 for normalization. There are a variety of practical reasons why standardizing the inputs can make training faster and reduce the chances of getting stuck in local optima. Normalization was done by using following equation;

$$U_{\text{Nor}} = \frac{U_{\text{actual}}}{U_{\max}}, \tag{4}$$

where U_{Nor} is the normalized value, U_{actual} is the actual value obtained from experiment, $U_{\max}$ is the maximum observation value of the data set. The normalized data set was then used as inputs to ANN.

Statistical parameters of ANN models are presented in Tables 8 and 9, where; R^2, MSE, and MAE correspond to the coefficient of correlation, mean square error and the mean absolute error. Table 8 shows the performances of the best ANN models for training phase. Table 9 shows the performances of the best ANN models for test data which consists of test and validation data. The patterns used in test and training sets are selected in systematic randomly. It seems that the network with 5 neurons in its hidden layer has better performance than the network with 4 neurons in its hidden layer for all training phases. On the other hand, though, the former has better performance for MMR and EWR test data, and the latter has showed better performance for R_a test data.

3.3. Development of Adaptive Nero-Fuzzy Inference System (ANFIS) Models. Fuzzy logic (FL) and neural networks (NN) supply two different methodologies to achieve an appropriate solution for uncertainty. Appropriate hybridization of fuzzy logic and neural networks technologies lead to overcoming the weakness of one with the strength of the other and so provide efficient solution to a range of problems belonging to different domains. A specific approach in Neuro-Fuzzy development is the Adaptive Neuro-Fuzzy Inference System (ANFIS) [29–31]. The ANFIS learning method works similarly to the neural networks' one. In ANFIS system, membership functions parameters are adjusted using either a backpropagation algorithm alone or in combination with a least squares type of method by using a given input/output data set. This adjustment allows your fuzzy systems to learn from the data from modelling [29].

Figure 5 shows a typical ANFIS architecture with two inputs (x and y), the linguistic labels (A1, A2, B1, and B2) associated with this node function, the normalized firing

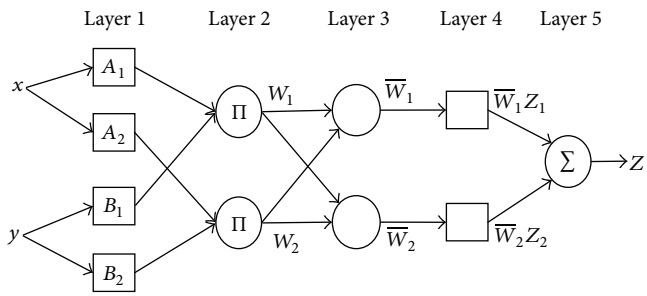


Figure 5: A typical ANFIS architecture [29].

Table 8: Statistical values of the best results of ANN modelling for training data.

Statistical parameters	MRR train data		EWR train data		R_a train data	
	4 neurons	5 neurons	4 neurons	5 neurons	4 neurons	5 neurons
MSE	11.0836	**3.4706**	0.58164	**0.01511**	0.01345	**0.00134**
MAE	2.2148	**1.3690**	0.52516	**0.10309**	0.08537	**0.02819**
R^2	0.9994	**0.9998**	0.99718	**0.99993**	0.99911	**0.99991**

Table 9: Statistical values of the best results of ANN modelling for test data.

Statistical parameters	MRR test data		EWR test data		R_a test data	
	4 neurons	5 neurons	4 neurons	5 neurons	4 neurons	5 neurons
MSE	28.2066	**27.7457**	1.99139	**1.99451**	**0.25138**	0.70232
MAE	3.4549	**3.7595**	0.91645	**0.96862**	**0.32167**	0.59253
R^2	0.9970	**0.9982**	0.99380	**0.99518**	**0.99404**	0.98739

strengths (W_i), and the node label (Π). ANFIS is a Sugeno-type fuzzy system in five-layered feed-forward network structure. A fuzzification process is performed in the first layer, the second layer is the rule layer, and each neuron in this layer corresponds to a single Sugeno-type fuzzy rule. The membership functions (MFs) are normalized in the third layer. The fourth layer is defuzzification layer where the consequent parts of the rules are executed. The fifth layer computes the overall ANFIS output as the summation of all input signals [32].

In this study, to build an ANFIS model and to predict the MRR, EWR, and R_a, the Adaptive Nero-Fuzzy Inference System (ANFIS) editor of MATLAB was used. The toolbox helps you to create fuzzy systems by using a backpropagation algorithm alone or in combination with a least squares method. Training and testing data set used in ANFIS models are the same as those used in ANN and GEP models. Here, testing data set was used as a checking data set. A checking data set is employed for verifying the ANFIS model generalization capabilities, except for training set. An initial FIS for training of ANFIS implementing subtractive clustering on the input/output data given is created by the model. A fast identification of parameters is obtained by using the hybrid learning algorithm so the necessary time to approach the convergence is reduced.

A membership function (MF) is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. The ANFIS toolbox includes 11 built-in membership function types: trimf, trapmf, gauss, bell, and so forth. The number of membership functions, MFs, and the type of input and output membership functions are chosen in the toolbox. In this study, due to their smoothness and concise notation, *Gaussian* and *bell* membership functions were chosen for inputs. The numbers of MFs were changed from 3-3-3 to 6-6-6. It means that $64 \times 2 = 128$ tries were carried out for each performance parameter. MF type was selected linear as the output membership function and epoch number was taken as 100.

Tables 10 and 11 show statistically the performances of the best ANFIS models that have Gaussian and bell membership functions for training and test phase, respectively. After training and testing, the number of MFs was fixed for each input variable, when the ANFIS model reaches the more acceptable satisfactory level at gauss MFs. For the best models, ANFIS architectures are illustrated in Table 12.

4. Results and Discussion

The results obtained from the modelling and experimental studies are compared in this section. Experimental data are illustrated experiment, and data obtained from modelling studies are illustrated as guess value at Tables 16–21. Training data sets for MRR, EWR, and R_a responses are given at Tables 16, 18, and 20 and testing data are given at Tables 17, 19, and 21,

Table 10: Statistical values of the best results of ANFIS modelling for training data.

Statistical parameters	MRR train data		EWR train data		R_a train data	
	GbellMF's (434)	GaussMF's (434)	GbellMF's (666)	GaussMF's (666)	GbellMF's (343)	GaussMF's (444)
MSE	37.4509	**1.9850**	0.0162	**1.9852**	0.0330	**0.0100**
MAE	3.0676	**0.7010**	2.8726	**0.7009**	0.1081	**0.0420**
R^2	0.9979	**0.9999**	0.9999	**0.9999**	0.9978	**0.9996**

Table 11: Statistical values of the best results of ANFIS modelling for testing data.

Statistical parameters	MRR test data		EWR test data		R_a test data	
	GbellMF's (434)	GaussMF's (434)	GbellMF's (666)	GaussMF's (666)	GbellMF's (343)	GaussMF's (444)
MSE	72.0726	**71.5620**	0.0013	**1.4000**	1.9228	**0.0670**
MAE	5.5172	**5.5080**	14.4491	**0.9860**	0.6852	**0.1780**
R^2	0.9944	**0.9944**	0.9632	**0.9961**	0.9738	**0.9973**

Table 12: ANFIS architecture and training parameters.

MFs type	MRR Gauss	EWR Gauss	R_a Gauss
Number of nodes	124	474	158
Number of linear parameters	192	864	256
Number of nonlinear parameters	22	36	36
Total number of parameters	214	900	292
Number of fuzzy rules	48	216	64
NumMFs input 1	4	6	4
NumMFs input 2	3	6	4
NumMFs input 3	4	6	4

Table 13: Statistical values of MRR modelling for training and testing sets.

Statistical parameters	MRR training sets			MRR test sets		
	GEP	ANN	ANFIS	GEP	ANN	ANFIS
MSE	55.352	3.4706	**1.985**	162.236	**27.7457**	71.562
MAE	5.486	1.3690	**0.701**	8.352	**3.7595**	5.508
R^2	0.9968	0.9998	**0.9999**	0.9958	**0.9982**	0.9944

Table 14: Statistical values of EWR modelling for training and testing sets.

Statistical parameters	EWR training sets			EWR test sets		
	GEP	ANN	ANFIS	GEP	ANN	ANFIS
MSE	3.550499	**0.01511**	1.985182	3.612	1.99451	**1.400**
MAE	1.296712	**0.10309**	0.700867	1.245	0.96862	**0.986**
R^2	0.98344	**0.99993**	0.999884	0.9885	0.99518	**0.9961**

Table 15: Statistical values of R_a modelling for training and testing sets.

Statistical Parameters	R_a Training Sets			R_a Test sets		
	GEP	ANN	ANFIS	GEP	ANN	ANFIS
MSE	0.1000	0.0100	**0.0100**	0.2200	0.25138	**0.0670**
MAE	0.2360	0.0850	**0.0420**	0.3380	0.32167	**0.1780**
R^2	0.9936	0.9991	**0.9996**	0.9942	0.99404	**0.9973**

Table 16: Training evaluation values of electrode wear rate (EWR).

Number	I (A)	T_{on} (μs)	T_{off} (μs)	Experiment	GEP		ANN		ANFIS	
					Guess	Ratio	Guess	Ratio	Guess	Ratio
1	3	3	3	15.00	18.68	0.80	15.0	1.00	15.0	1.00
2	3	12	3	4.00	5.25	0.76	3.9	1.02	4.0	1.00
3	3	25	6	1.30	1.76	0.74	1.2	1.06	1.3	1.00
4	3	100	24	0.50	0.15	3.32	0.4	1.32	0.5	1.00
5	6	6	3	28.00	22.40	1.25	27.8	1.01	27.9	1.00
6	6	12	3	15.00	13.51	1.11	15.1	0.99	15.1	0.99
7	6	24	6	6.00	6.53	0.92	6.1	0.98	5.9	1.02
8	6	48	6	2.00	2.61	0.77	2.1	0.97	2.0	1.00
9	6	99	12	0.80	0.84	0.95	0.6	1.38	0.8	1.00
10	12.5	3	3	38.00	37.67	1.01	38.2	1.00	38.0	1.00
11	12.5	12	3	24.00	23.88	1.00	23.9	1.01	24.0	1.00
12	12.5	24	6	13.00	15.02	0.87	12.9	1.00	13.0	1.00
13	12.5	48	6	4.80	8.04	0.60	4.8	0.99	4.9	0.98
14	12.5	201	24	0.10	1.30	0.08	0.3	0.33	0.1	1.00
15	12.5	402	48	0.10	0.43	0.23	0.1	1.16	0.1	1.00
16	25	3	3	40.00	42.06	0.95	39.9	1.00	40.0	1.00
17	25	6	3	38.00	38.57	0.99	38.2	1.00	38.1	1.00
18	25	24	6	24.00	24.00	1.00	24.0	1.00	23.9	1.00
19	25	48	6	14.00	15.59	0.90	13.9	1.01	14.1	0.99
20	25	99	12	6.00	8.54	0.70	6.2	0.96	6.0	1.00
21	25	201	24	2.20	4.11	0.54	2.1	1.06	2.2	1.00
22	25	801	99	0.10	0.65	0.15	0.1	1.30	0.1	1.00
23	50	3	3	45.00	44.52	1.01	44.8	1.00	45.0	1.00
24	50	12	3	38.00	38.50	0.99	38.2	1.00	38.0	1.00
25	50	24	6	32.00	32.12	1.00	32.0	1.00	32.0	1.00
26	50	48	6	25.00	24.08	1.04	24.9	1.00	25.0	1.00
27	50	99	12	20.00	15.53	1.29	20.1	1.00	20.0	1.00
28	50	201	24	10.00	8.88	1.13	9.9	1.01	10.0	1.00
29	50	402	48	4.50	4.59	0.98	4.6	0.98	4.5	1.00
30	50	801	99	1.40	2.14	0.65	1.4	1.00	1.4	1.00

Table 17: Test values of electrode wear ratio (EWR).

Number	I (A)	T_{on} (μs)	T_{off} (μs)	Experiment	GEP		ANN		ANFIS	
					Guess	Ratio	Guess	Ratio	Guess	Ratio
1	3	6	3	12.00	11.54	1.04	9.67	1.24	11.84	1.01
2	3	50	12	0.50	0.54	0.93	0.37	1.35	0.87	0.58
3	6	3	3	35.00	29.75	1.18	32.94	1.06	33.03	1.06
4	6	201	24	0.10	0.24	0.41	0.25	0.39	0.10	0.98
5	12.5	6	3	33.00	32.21	1.02	33.43	0.99	34.16	0.97
6	12.5	99	12	1.70	3.49	0.49	1.47	1.16	2.25	0.75
7	25	12	3	31.00	32.32	0.96	32.22	0.96	32.63	0.95
8	25	402	48	0.40	1.75	0.23	0.21	1.91	1.53	0.26
9	50	6	3	42.00	42.51	0.99	41.58	1.01	43.11	0.97
10	50	801	99	0.01	0.89	0.11	0.18	0.54	1.88	0.05

Table 18: Training evaluation values of material removal rate (MRR) (mm^3/min) for all modelling method.

Number	I (A)	T_{on} (μs)	T_{off} (μs)	Experiment	GEP		ANN		ANFIS	
					Guess	Ratio	Guess	Ratio	Guess	Ratio
1	3	3	3	3.40	0.20	17.36	3.2	1.07	3.6	0.94
2	3	12	3	7.00	−1.57	−4.46	7.0	1.01	6.9	1.01
3	3	24	6	8.00	−2.72	−2.94	9.6	0.83	7.5	1.06
4	3	99	24	6.50	−4.49	−1.45	6.0	1.08	6.5	1.00
5	6	6	3	14.00	14.77	0.95	13.8	1.02	13.2	1.06
6	6	12	3	20.00	18.27	1.09	18.0	1.11	20.2	0.99
7	6	24	6	22.00	20.87	1.05	22.7	0.97	22.5	0.98
8	6	48	6	19.00	13.77	1.38	18.4	1.04	19.3	0.98
9	6	99	12	12.00	7.92	1.51	12.9	0.93	11.9	1.01
10	12.5	3	3	26.00	29.99	0.87	33.3	0.78	26.4	0.98
11	12.5	12	3	53.00	56.03	0.95	53.4	0.99	52.7	1.01
12	12.5	24	6	67.00	66.55	1.01	65.2	1.03	67.1	1.00
13	12.5	48	6	70.00	68.67	1.02	65.5	1.07	69.8	1.00
14	12.5	201	24	55.00	61.07	0.90	56.9	0.97	55.0	1.00
15	12.5	402	48	52.00	56.08	0.93	51.9	1.00	51.9	1.00
16	25	3	3	70.00	67.32	1.04	69.2	1.01	74.5	0.94
17	25	6	3	90.00	94.67	0.95	85.7	1.05	86.2	1.04
18	25	24	6	150.00	149.52	1.00	150.5	1.00	149.8	1.00
19	25	48	6	150.00	164.47	0.91	152.9	0.98	149.8	1.00
20	25	99	12	150.00	170.27	0.88	152.9	0.98	150.5	1.00
21	25	201	24	155.00	168.54	0.92	153.0	1.01	154.8	1.00
22	25	801	99	155.00	159.85	0.97	154.8	1.00	154.8	1.00
23	50	3	3	150.00	141.26	1.06	150.0	1.00	153.3	0.98
24	50	12	3	255.00	258.10	0.99	256.1	1.00	251.6	1.01
25	50	24	6	310.00	311.33	1.00	308.2	1.01	310.1	1.00
26	50	48	6	350.00	347.83	1.01	360.4	0.97	350.2	1.00
27	50	99	12	370.00	368.35	1.00	365.8	1.01	370.1	1.00
28	50	201	24	390.00	375.10	1.04	375.8	1.04	390.0	1.00
29	50	402	48	380.00	373.81	1.02	387.9	0.98	380.3	1.00
30	50	801	99	370.00	369.84	1.00	370.3	1.00	370.1	1.00

Table 19: Test values of material removal rate (MRR) (mm^3/min).

Number	I (A)	T_{on} (μs)	T_{off} (μs)	Experiment	GEP		ANN		ANFIS	
					Guess	Ratio	Guess	Ratio	Guess	Ratio
1	3	6	3	5.00	0.61	8.21	4.98	1.00	5.07	0.99
2	3	48	12	8.00	−3.71	−2.08	8.31	0.93	13.53	0.57
3	6	3	3	10.00	10.01	1.00	10.55	0.95	10.88	0.92
4	6	201	24	7.00	2.46	2.85	5.59	1.25	3.36	2.08
5	12.5	6	3	36.00	42.84	0.84	41.17	0.87	36.97	0.97
6	12.5	99	12	62.00	66.36	0.93	61.98	1.00	58.50	1.06
7	25	12	3	130.00	124.34	1.05	114.71	1.13	122.46	1.06
8	25	402	48	155.00	163.70	0.95	154.08	1.01	177.06	0.88
9	50	6	3	195.00	196.89	0.99	183.70	1.06	205.53	0.95
10	50	801	99	330.00	365.71	0.90	330.23	1.00	329.94	1.00

Table 20: Training values of surface roughness (R_a) (μm).

Number	I (A)	T_{on} (μs)	T_{off} (μs)	Experiment	GEP		ANN		ANFIS	
					Guess	Ratio	Guess	Ratio	Guess	Ratio
1	3	3	3	1.40	0.89	1.57	1.40	1.00	1.39	1.01
2	3	12	3	2.00	1.80	1.11	1.96	1.02	2.00	1.00
3	3	24	6	2.20	1.84	1.20	2.22	0.99	2.21	1.00
4	3	99	24	3.20	3.24	0.99	3.20	1.00	3.20	1.00
5	6	6	3	2.00	2.18	0.92	2.00	1.00	2.02	0.99
6	6	12	3	2.50	2.74	0.91	2.45	1.02	2.50	1.00
7	6	24	6	2.80	3.04	0.92	2.84	0.99	2.79	1.00
8	6	48	6	3.50	3.82	0.92	3.44	1.02	3.49	1.00
9	6	99	12	4.00	4.20	0.95	4.05	0.99	4.00	1.00
10	12.5	3	3	2.20	2.19	1.01	2.20	1.00	2.32	0.95
11	12.5	12	3	4.00	4.16	0.96	4.00	1.00	4.09	0.98
12	12.5	24	6	5.00	4.95	1.01	4.97	1.01	4.99	1.00
13	12.5	48	6	5.60	5.89	0.95	5.65	0.99	5.94	0.94
14	12.5	201	24	8.00	7.48	1.07	7.98	1.00	7.99	1.00
15	12.5	402	48	9.00	9.07	0.99	9.00	1.00	9.00	1.00
16	25	3	3	3.20	3.30	0.97	3.28	0.97	3.28	0.98
17	25	6	3	4.00	4.15	0.96	3.94	1.01	3.92	1.02
18	25	24	6	5.60	5.89	0.95	5.62	1.00	5.63	0.99
19	25	48	6	7.00	7.00	1.00	6.99	1.00	7.00	1.00
20	25	99	12	8.00	8.39	0.95	7.97	1.00	7.99	1.00
21	25	201	24	10.00	9.95	1.00	10.02	1.00	9.85	1.02
22	25	801	99	12.60	13.21	0.95	12.60	1.00	12.60	1.00
23	50	3	3	4.00	4.01	1.00	3.96	1.01	4.05	0.99
24	50	12	3	6.30	5.89	1.07	6.35	0.99	6.32	1.00
25	50	24	6	7.00	7.00	1.00	6.98	1.00	7.02	1.00
26	50	48	6	9.00	8.32	1.08	8.95	1.01	8.98	1.00
27	50	99	12	10.00	9.97	1.00	10.07	0.99	10.12	0.99
28	50	201	24	12.60	11.91	1.06	12.56	1.00	12.60	1.00
29	50	402	48	14.00	14.16	0.99	14.00	1.00	14.05	1.00
30	50	801	99	16.00	15.88	1.01	16.00	1.00	16.00	1.00

Table 21: Test values of surface roughness (R_a) (μm).

Number	I (A)	T_{on} (μs)	T_{off} (μs)	Experiment	GEP		ANN		ANFIS	
					Guess	Ratio	Guess	Ratio	Guess	Ratio
1	3	6	3	1.6	1.33	1.20	1.5	1.05	1.62	0.99
2	3	50	12	2.8	2.45	1.14	2.8	1.01	2.30	1.22
3	6	3	3	1.6	1.70	0.94	1.6	1.00	1.71	0.94
4	6	201	24	4.5	5.34	0.84	5.2	0.86	4.97	0.91
5	12.5	6	3	3.2	3.48	0.92	3.0	1.05	3.02	1.06
6	12.5	99	12	7	6.87	1.02	6.3	1.11	6.62	1.06
7	25	12	3	5	4.95	1.01	5.1	0.98	4.95	1.01
8	25	402	48	11.2	10.97	1.02	11.1	1.01	11.20	1.00
9	50	6	3	5	4.94	1.01	4.9	1.03	4.930	1.01
10	50	1602	201	18	19.07	0.94	16.8	1.07	18.000	1.00

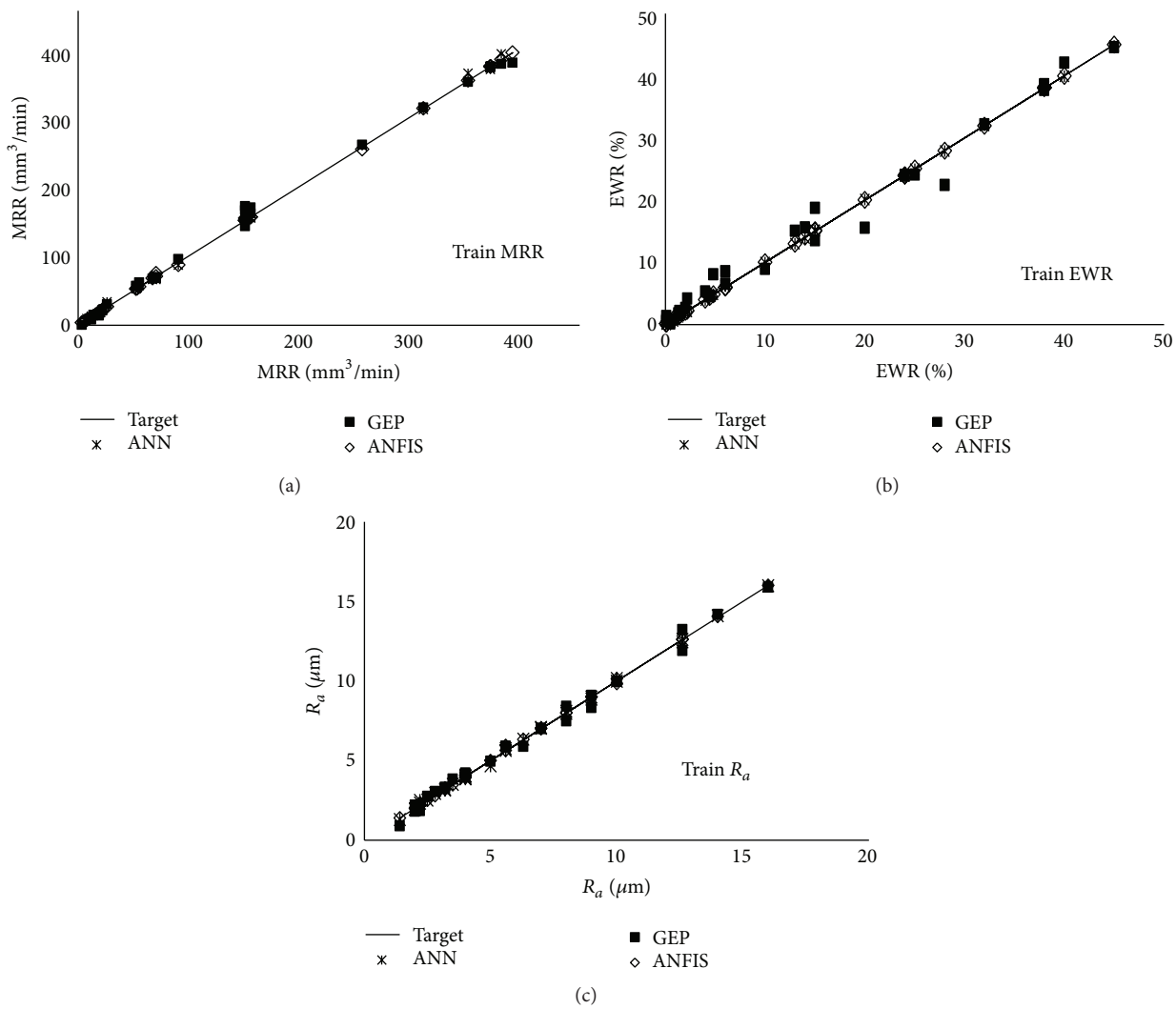


Figure 6: Comparisons of training performance of all methods for (a) MRR, (b) EWR, and (c) R_a prediction.

respectively. Ratio means rate of the experimental data to the predicted value obtained from modelling study. Estimation capabilities of all artificial models for each experimental data can be evaluated as quantitatively with the help of ratio values.

Comparisons of the training performances and testing performances of all models with the experimental equivalents are given as graphically in Figures 6 and 7. As shown in these figures, there is a little deviation between the real and predicted values. It means that all of these models can predict accurately and satisfactorily EDM performances by learning complex relationship between the input and output parameters.

MSE, MAE, and R^2 values are used to compare statistically the created models. Statistical performances of MRR, EWR, and R_a modelling for training and testing sets are given at Tables 13, 14, and 15, respectively. Although the ANFIS model is very successful in training (R^2 = 0.9999, MAE = 0.701, and MSE = 1.985), the ANN model is better in prediction of MRR (R^2 = 0.9982, MAE = 3.7595, and MSE = 27.7457) (Table 13). For EWR modelling, the ANN model is more successful in training than the ANFIS model but the ANFIS model is slightly better in prediction of EWR than the ANN model (Table 14). The ANFIS model is the best for training and prediction of R_a in all models (Table 15).

GEP model is less successful compared with other models. In prediction of EWR and R_a, the ANFIS model is the best as shown in Tables 13 and 14. All models are better in prediction of R_a than MRR. The differences between the ANN and the ANFIS results in terms of R^2, MSE, and MAE, are very close, so that both models may be accepted as successful. However, the ANFIS model is slightly better than the ANN and the GEP model.

5. Conclusion

In this paper, a comprehensive study to compare the performances of GEP, ANN, and ANFIS approaches on modelling of important EDM performances was realized. Three basic parameters discharge current (I), pulse on-time (T_{on}), and pulse off-time (T_{off}) were input parameters while material

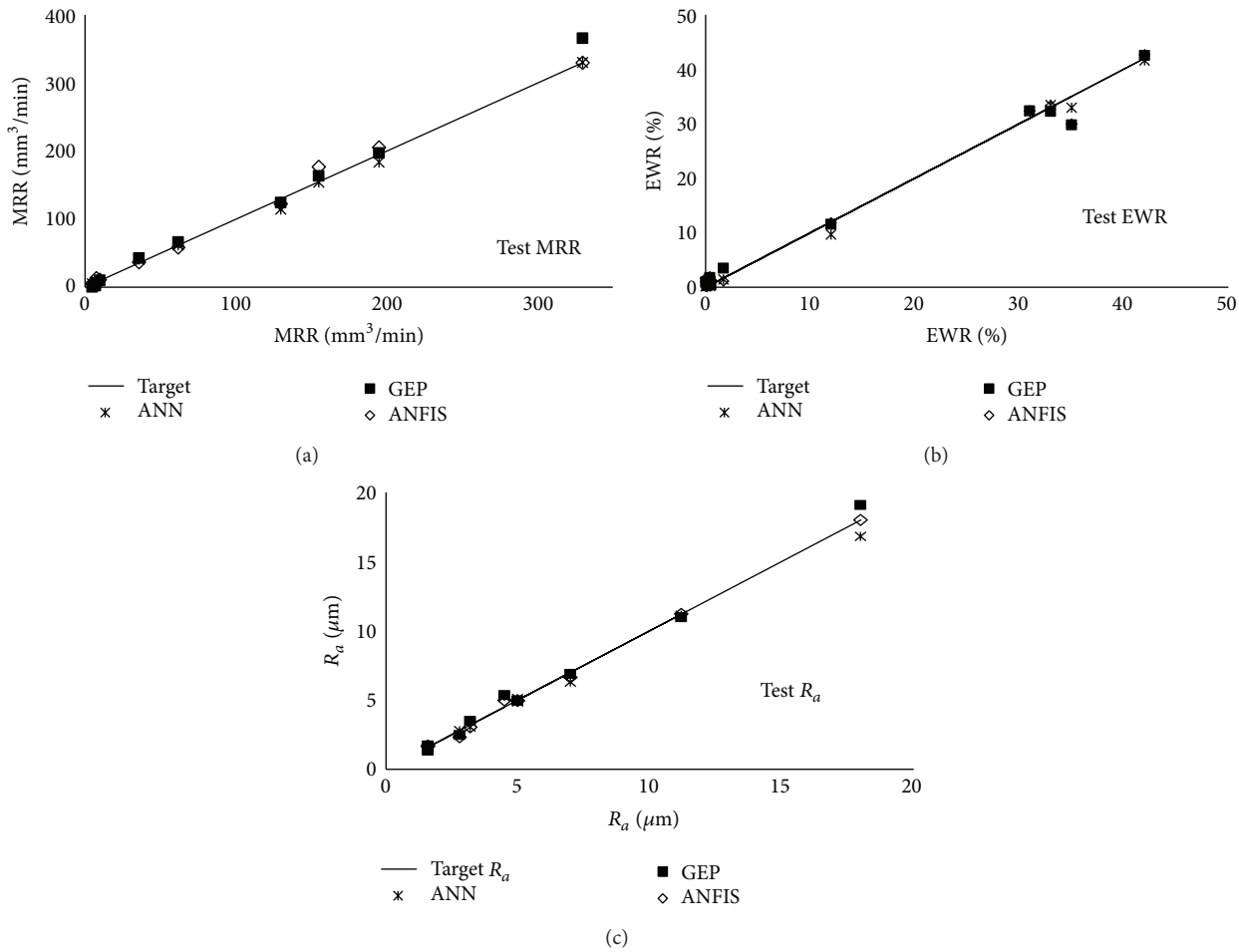


Figure 7: Comparisons of testing performance of all methods for (a) MRR, (b) EWR, and (c) R_a prediction.

removing rate (MRR), electrode wear ratio (EWR), and surface roughness (R_a) were the output parameters. A series of experiments was carried out using DIN 1.2080 tool steel and copper electrode. The data set obtained from experimental study was used to develop mathematical models.

(i) The results show that all approaches are successful for prediction of EDM performances by learning complex relationship between the input and output parameters in terms of statistical parameters.

(ii) GEP model was less successful among others; however, it has a simple and explicit mathematical formulation.

(iii) The results produced by both ANN and ANFIS models also present relatively good level of accuracy. However, the ANFIS model is slightly more successful than the ANN model.

(iv) Modeling of R_a is more efficient compared to other EDM performances.

(v) Using of MLP with back-propagation learning algorithm (while it has one hidden layer, TanhAxon, and 5 neurons) is successful for modelling EDM performances.

(vi) Selection of gauss MFs type gives better results than the others MFs types for modelling of EDM performances by using ANFIS method.

Acknowledgment

The authors would like to thank the Scientific Research Projects (BAP) Unit of Gaziantep University for supporting this study.

References

[1] C. Wang, H. Chow, L. Yang, and C. Lu, "Recast layer removal after electrical discharge machining via Taguchi analysis: a feasibility study," *Journal of Materials Processing Technology*, vol. 209, no. 8, pp. 4134–4140, 2009.

[2] D. Mandal, S. K. Pal, and P. Saha, "Modeling of electrical discharge machining process using back propagation neural network and multi-objective optimization using non-dominating sorting genetic algorithm—II," *Journal of Materials Processing Technology*, vol. 186, no. 1-3, pp. 154–162, 2007.

[3] S. H. Lee and X. P. Li, "Study of the effect of machining parameters on the machining characteristics in electrical discharge

machining of tungsten carbide," *Journal of Materials Processing Technology*, vol. 115, no. 3, pp. 344–358, 2001.

[4] N. K. Jain and V. K. Jain, "Modeling of material removal in mechanical type advanced machining processes: a state-of-art review," *International Journal of Machine Tools and Manufacture*, vol. 41, no. 11, pp. 1573–1635, 2001.

[5] N. Mohd Abbas, D. G. Solomon, and M. Fuad Bahari, "A review on current research trends in electrical discharge machining (EDM)," *International Journal of Machine Tools and Manufacture*, vol. 47, no. 7-8, pp. 1214–1228, 2007.

[6] G. Indurkhya and K. P. Rajurkar, "Artificial Neural Network approach in modelling of EDM process," in *Proceedings of the Artificial Neural Networks in Engineering (ANNIE '92)*, pp. 845–850, November 1992.

[7] K. Tsai and P. Wang, "Comparisons of neural network models on material removal rate in electrical discharge machining," *Journal of Materials Processing Technology*, vol. 117, no. 1-2, pp. 111–124, 2001.

[8] K. M. Tsai and P. J. Wang, "Predictions on surface finish in electrical discharge machining based upon neural network models," *International Journal of Machine Tools and Manufacture*, vol. 41, no. 10, pp. 1385–1403, 2001.

[9] C. L. Lin, J. L. Lin, and T. C. Ko, "Optimisation of the EDM process based on the orthogonal array with fuzzy logic and grey relational analysis method," *International Journal of Advanced Manufacturing Technology*, vol. 19, no. 4, pp. 271–277, 2002.

[10] K. Wang, H. L. Gelgele, Y. Wang, Q. Yuan, and M. Fang, "A hybrid intelligent method for modelling the EDM process," *International Journal of Machine Tools and Manufacture*, vol. 43, no. 10, pp. 995–999, 2003.

[11] D. K. Panda and R. K. Bhoi, "Artificial neural network prediction of material removal rate in electro discharge machining," *Materials and Manufacturing Processes*, vol. 20, no. 4, pp. 645–672, 2005.

[12] O. Yilmaz, O. Eyercioglu, and N. N. Z. Gindy, "A user-friendly fuzzy-based system for the selection of electro discharge machining process parameters," *Journal of Materials Processing Technology*, vol. 172, no. 3, pp. 363–371, 2006.

[13] Ö. Salman and M. C. Kayacan, "Evolutionary programming method for modeling the EDM parameters for roughness," *Journal of Materials Processing Technology*, vol. 200, no. 1–3, pp. 347–355, 2008.

[14] G. K. M. Rao, G. R. Janardhana, D. H. Rao, and M. S. Rao, "Development of hybrid model and optimization of surface roughness in electric discharge machining using artificial neural networks and genetic algorithm," *Journal of Materials Processing Technology*, vol. 209, no. 3, pp. 1512–1520, 2009.

[15] K. Maji and D. K. Pratihar, "Forward and reverse mappings of electrical discharge machining process using adaptive network-based fuzzy inference system," *Expert Systems with Applications*, vol. 37, no. 12, pp. 8566–8574, 2010.

[16] S. N. Joshi and S. S. Pande, "Intelligent process modeling and optimization of die-sinking electric discharge machining," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 2743–2755, 2011.

[17] K. H. Ho and S. T. Newman, "State of the art electrical discharge machining (EDM)," *International Journal of Machine Tools and Manufacture*, vol. 43, no. 13, pp. 1287–1300, 2003.

[18] M. Kunieda, B. Lauwers, K. P. Rajurkar, and B. M. Schumacher, "Advancing EDM through fundamental insight into the process," *CIRP Annals*, vol. 54, no. 2, pp. 599–622, 2005.

[19] Y. H. Guu, C. Y. Chou, and S. Chiou, "Study of the effect of machining parameters on the machining characteristics in electrical discharge machining of Fe-Mn-Al alloy," *Materials and Manufacturing Processes*, vol. 20, no. 6, pp. 905–916, 2005.

[20] C. S. Ong, J. J. Huang, and G. H. Tzeng, "Building credit scoring models using genetic programming," *Expert Systems with Applications*, vol. 29, no. 1, pp. 41–47, 2005.

[21] A. Cevik and I. H. Guzelbey, "A soft computing based approach for the prediction of ultimate strength of metal plates in compression," *Engineering Structures*, vol. 29, no. 3, pp. 383–394, 2007.

[22] M. Kök, E. Kanca, and Ö. Eyercioğlu, "Prediction of surface roughness in abrasive waterjet machining of particle reinforced MMCs using genetic expression programming," *International Journal of Advanced Manufacturing Technology*, vol. 55, no. 9–12, pp. 955–968, 2011.

[23] J. R. Koza, "Genetic programming as a means for programming computers by natural selection," *Statistics and Computing*, vol. 4, no. 2, pp. 87–112, 1994.

[24] C. Ferreira, "Gene expression programming: a new adaptive algorithm for solving problems," *Complex Systems*, vol. 13, no. 2, pp. 87–129, 2001.

[25] A. Cevik, "Genetic programming based formulation of rotation capacity of wide flange beams," *Journal of Constructional Steel Research*, vol. 63, no. 7, pp. 884–893, 2007.

[26] S. Haykin, *Neural Networks and Learning Machines*, Prentice Hall, Upper Saddle River, NJ, USA, 3rd edition, 2009.

[27] M. Y. Rafiq, G. Bugmann, and D. J. Easterbrook, "Neural network design for engineering applications," *Computers and Structures*, vol. 79, no. 17, pp. 1541–1552, 2001.

[28] I. H. Guzelbey, A. Cevik, and A. Erklig, "Prediction of web crippling strength of cold-formed steel sheetings using neural networks," *Journal of Constructional Steel Research*, vol. 62, no. 10, pp. 962–973, 2006.

[29] J. S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.

[30] J. S. R. Jang, T. C. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing: A computational Approach to Learning and Machine Intelligence*, Prentice Hall, 1997.

[31] J. Kim and N. Kasabov, "HyFIS: adaptive neuro-fuzzy inference systems and their application to nonlinear dynamical systems," *Neural Networks*, vol. 12, no. 9, pp. 1301–1319, 1999.

[32] M. Sugeno and G. T. Kang, "Structure identification of fuzzy model," *Fuzzy Sets and Systems*, vol. 28, no. 1, pp. 15–33, 1988.

Hindawi Publishing Corporation
Advances in Mechanical Engineering
Volume 2013, Article ID 628027, 6 pages
http://dx.doi.org/10.1155/2013/628027

Research Article

Mechanism and Motion of Semifixed Abrasive Grit for Wire-Saw Slicing

Chunyan Yao, Wei Peng, and Siyuan Chen

Key Laboratory of E&M, Zhejiang University of Technology, Ministry of Education & Zhejiang Province, Hangzhou 310023, China

Correspondence should be addressed to Wei Peng; pengwei@zjut.edu.cn

Received 18 April 2013; Accepted 9 May 2013

Academic Editor: Shengyong Chen

The currently dominant method in the production of wafers is to use slurry wire-saw slicing. This paper reports a new wire-saw slicing technology, namely, semifixed abrasive wire-saw slicing. The traditional smooth wire is replaced by a patterned wire with a textured surface that can help the wire carrying the abrasives. It aims to improve the number of transient fixed abrasives in the machining process. Transient fixed abrasives can produce "scratch-indenting" processes that are similar to fixed-abrasive wire slicing thereby improving the efficiency of the wire-saw cutting. This study focuses on the behavioral mechanics of the abrasive grits in the slurry during the slicing process. The dynamic images of the movement of abrasive grits in the slicing process are obtained by a high-speed camera. At the microlevel, using statistical methods, the behavioral mechanics of the abrasive grits are investigated by changing the slicing parameters. The results contribute toward a more intuitive and profound understanding of the principle of free-abrasive wire sawing.

1. Introduction

Wire-saw slicing technology has conventionally been used for cutting monocrystalline and polycrystalline silicon ingots [1–6]. The mainstream technology currently adopted is free-abrasive slurry wire-saw slicing. In the slicing process of the free-abrasive wire-saw, material is continuously removed through the interaction of the abrasive particles below the moving wire and the silicon surface. Different behavioral mechanics of the abrasive grits in the slurry determine the mechanism of material removal. Thickness uniformity of wafers is a critical quality measure in a wire-saw slicing process. Nonuniformity occurs when the material removal rate changes over time during a slicing process, and it poses a significant problem for the downstream processes such as lapping and polishing [7]. A partial differential equation-constrained Gaussian process model is developed by Zhao et al. by global Galerkin discretization with three features incorporated into the statistical model [8].

So far, studies on the mechanism of free-abrasive wire saw have made some important advances. The earlier work done by Bhagavat and Kao described the removal mechanism during the wire-sawing process as a "rolling-indenting" model with three-body abrasion [3, 4, 9]. In this case, rolling abrasives between the moving wire and the workpiece are randomly rotated and indented into the surface under film pressure to generate cracks and make the material chip away from the substrate. Based on the rolling, indenting, and scratching modes of the free abrasives, slicing models of "rolling-indenting" and "scratch-indenting" were indicated by Yang [10]. Bhagavat et al. studied hydrodynamic pressure in the slicing area, and the "rolling-indenting" model was verified with the observation of the surface morphology of the silicon wafer produced [11].

Through a comprehensive research of cutting mechanisms and abrasive movement states of multiwire sawing, Möller et al. pointed out the abrasive particles as freely dispersed in the cutting zone [12], which generate both a "semicontact" and "noncontact" case as illustrated in Figure 1. In the first case, the positive pressure normal to the surface and the shear stress parallel to it on the grains are exerted by the moving wire directly to rotate and indent into the surface, and the "rolling-indenting" mode is mainly the grinding pattern. In the second case, the force on the grits is supplied by the shear stress alone in the moving slurry to create a rollover effect.

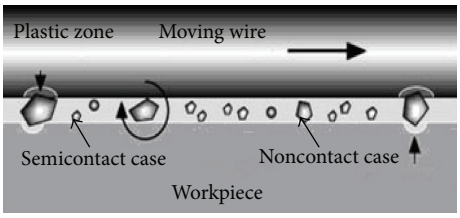


Figure 1: Schematic diagram of the semicontact case and the noncontact case.

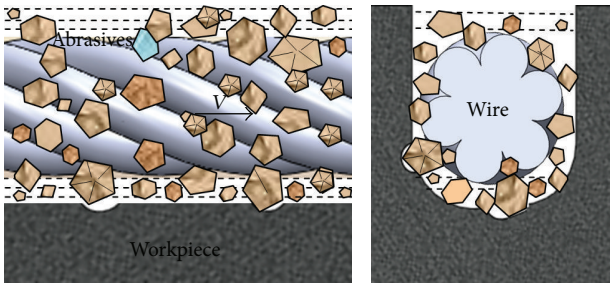


Figure 2: Schematic of semifixed wire sawing.

Besides, Ge found similar results through the research and its mechanism of large-diameter silicon ingot precise slicing with low damage [13]. Cheng et al. proposed a "rolling-indenting" and "rolling-indenting-chipping" model with multiple movement states [14]. The main results of the experimental investigations suggested that the particles in the semicontact case would remove the material more efficiently, whereas, in the noncontact case, they only rotate without any efficient energy transmission to the surface [11, 12]. Thus, a novel semifixed abrasive wire-saw slicing technology, in which the wire is characterized typically by a surface texture that will expand the quantity of semicontact abrasives to improve the efficiency in the process of machining, is proposed.

Worldwide research on cutting mechanisms and abrasive movement states of multiwire sawing is based on assumptions and experimental verification [15–17]; the motion state cannot be revealed directly during the process, and, thereby, the cutting mechanisms are essentially studied by means of microscopic observations of the moving abrasive state in semifixed abrasive wire-saw slicing. In this paper, all the experiments are observed using high-speed camera equipment, and the dynamic images of the moving-grits state in the cutting process are obtained and the motion properties analyzed by computers. Analysis of the motion state of abrasive particles in the cutting process in terms of different process parameters and different wire patterns helps in the intuitive and profound grasp of the cutting principle involved in free-abrasive wire sawing [18–20].

2. Semifixed Abrasive Wire-Saw Slicing

Studies have discovered that surface material removal mainly depends on the semicontact particles driven by the moving wire in efficient cutting by the "rolling-indenting" mode, whereas the non-contact ones simply involve rolling and crashing. Therefore, a new kind of slicing method termed semifixed abrasive wire-saw slicing has been put forward, which is novel compared to the traditional free-abrasive wire sawing, wherein the wire appearance is reformed or rather the special surface microgroove structures lead to more particles into the semicontact state. By heading the cutting region, the particles are adhered to wire surfaces in the dual action of fluted geometry tessellation and slurry hydrodynamic effect, and the transient fixed abrasives emerge incidentally, referred to as semifixed abrasive state, as illustrated in Figure 2. It is these channels, in favor of turning more abrasive particles into the semifixed state, that make the abrasives embedded in the microgroove and no longer show rolling motion but produce "scratch-indenting" processes that are similar to fixed-abrasive wire slicing, thereby improving the efficiency of wire-saw cutting. In the experiment, a multistrand wire with naturally formed grooved patterns with individual strands of the same diameter is used as a semifixed abrasive sawing wire.

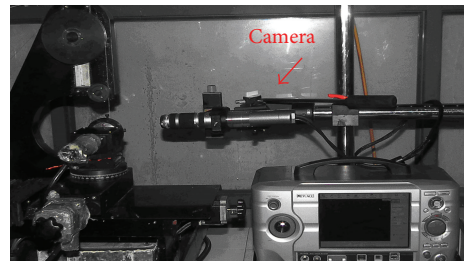


Figure 3: Observation platform of the abrasive.

3. System Design

3.1. Experimental Observation Platform. The motion states of abrasive particles of free-abrasive and semifixed abrasive wire sawing were accurately observed and recorded in real time by the use of the Keyence dynamic three-dimensional digital microscope VW-6000/5000 and were analyzed by computer programs. WXD170 reciprocating wire-saw machine was also used in the simulated cutting experiment. The entire observation platform is depicted in Figure 3.

3.2. Experimental Conditions. For observations of the abrasive grits' movements, the cutting parameters in simulation are different from those in reality, and the high-speed digital camera was set at the rate of 1000 frames per second (fps) with a magnification of 200× in the experiment. In addition, the shooting was carried out from the kerf side to enhance the viewing effect. As in the experimental conditions given in Table 1, mixed abrasive slurries were prepared by mixing black corundum abrasives and PEG300 at a proportion of 1 : 20. All sawing wires had the same diameter of 0.5 mm, including an individual strand called wire A, 1×7 multistrand wire called wire B, and 7×7 multistrand wire called wire C. Meanwhile, the pregrooved optical glass K9 was adopted as the workpiece.

Table 1: Slicing conditions in the experiment.

Experimental parameter	Value
Feed velocity (m/s)	0.25, 0.5
Initial tension (MPa)	0.2
Abrasive grits (mesh)	$170^{\#}$
Workpiece dimension (mm)	$30 \times 30 \times 70$
Wire diameter	Ø 0.5

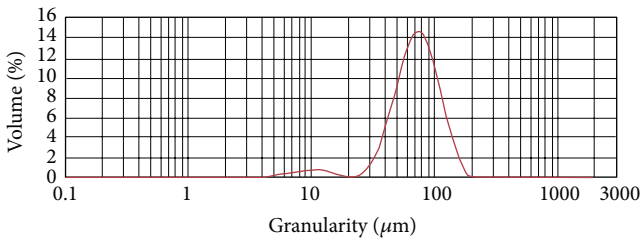


Figure 4: Grain size distribution.

3.3. Granulometry of Abrasive Particles. Owing to the purchased abrasive particle having a very extensive grain size distribution, sieving was done first for the determination of grain size and then the granularity was measured by a Malvern Mastersizer 2000 laser particle-size analyzer. The experimental abrasives have a mean size of 87 μm. Figure 4 gives the complete distribution.

3.4. Velocity of Moving Abrasives. Although the grits' movement was triggered by the joint action of the moving wire and abrasive suspension, the high-speed camera equipment does not actually move during photography. The real-time continuous sequence of images was analyzed, and the adjacent marked points' displacements were measured to get the net slip of a single grit. The average velocity of the particle was computed by the time lag between adjacent frames identified using the camera shooting parameters.

4. Experiments and Analysis

4.1. Abrasive Motion State in Semifixed Abrasive Slicing. During the semifixed abrasive wire-saw slicing, which is analogous to the free-abrasive wire sawing, the cutting area between the sawing wire and the workpiece is filled with mixed slurry driven by the moving taut wire to cut the workpiece. By the film formed with sufficient slurry swarming into the processing region, the wire appears to undergo elastic deformation and floats freely. With the wire running in high-speed feed, the slurry generates some hydrodynamic effect. Thus, the general mechanical behavior depends, to a considerable extent, on the hydrodynamic behavior of the abrasive suspension film in the cutting zone [11]. Given the diverse film thicknesses and abrasive sizes, advancement of the abrasive comes in different forms of state probabilities. The abrasive motion state is plentifully observed in real time during the wire-sawing processes. Based on the abrasive's own movement and in combination with the moving wire, the motion state can be classified into essentially two categories as the semifixed state and the free-rolling state. Abrasives in the former state are in closer contact with both the wire and the workpiece, playing a scratching role, which is similar to plastic cutting deformation. However, in the latter state, like a roller coaster, crashing with the wire and the workpiece and suspending in the slurry occasionally, abrasives thrive on the hydrodynamic effect to rotate and indent the workpiece surface.

4.2. Abrasive Motion Analysis in the Cutting Process. As mentioned above, the use of a dynamic three-dimensional digital camera during the simulated process allows easy observations of various abrasive movement states. Depending on the film thicknesses between the wire and the workpiece, the particles are either in direct contact with both the wire and the workpiece (semicontact case) or the particles are floating freely (noncontact case). As Figure 5(a) illustrates, three frames are continuously captured from the video image sequences, and it can be seen that an abrasive can readily get in touch with the moving wire and the workpiece simultaneously when the slurry film thickness is close to that of the abrasive particle diameter. The abrasives remain at the bottom of the cutting region, away from being attached to the work surface, and are partially embedded onto the moving wire leading to the removal of the workpiece material through an action similar to "ploughing," and then the semifixed abrasives (semifixed state I) emerge. Figure 5(b) shows another facet of the semifixed state. The images are captured in every other frame during the process. It is also evident that a bigger grit has fixed itself onto the surface microgroove, rolled along it, and slid into the flank of the cutting area. When the abrasive particle's diameter is greater than that of the slurry film thickness, it will produce the cutting effect by making contact with the workpiece. Otherwise, it just rolls through the abrasive suspension. It can also be seen in Figure 5(c) that the typical abrasive movement state in both the free- and semifixed abrasive wire-sawing processes is rolling characterized by a contact between the moving wire and the workpiece that is more abrasive. It is evident that, owing to the existence of the microgroove, more abrasives are transformed as transient fixed abrasives, that is, "semifixed abrasive," to scratch the surface material and improve the efficiency of the wire-saw cutting process. Correspondingly, however, the slicing wire in the conventional free-abrasive wire sawing has no existing microgroove or surface structure, and, therefore, most of the abrasive particles collide back and forth during material removal under the actions of the moving wire and slurry hydrodynamics.

4.3. Abrasive Motion State Statistical Analysis in the Cutting Process. The abrasive movement state of the cutting process is visualized in a portion of the screenshots and then, considering the current practice, classified and analyzed, using statistical methods, in the computer by means of handling a video. Figure 6 gives a statistical analysis graph of different abrasive motion states in the central cutting area, showing the diverse movements of the sawing wires at a feed rate of 0.25 m/s.

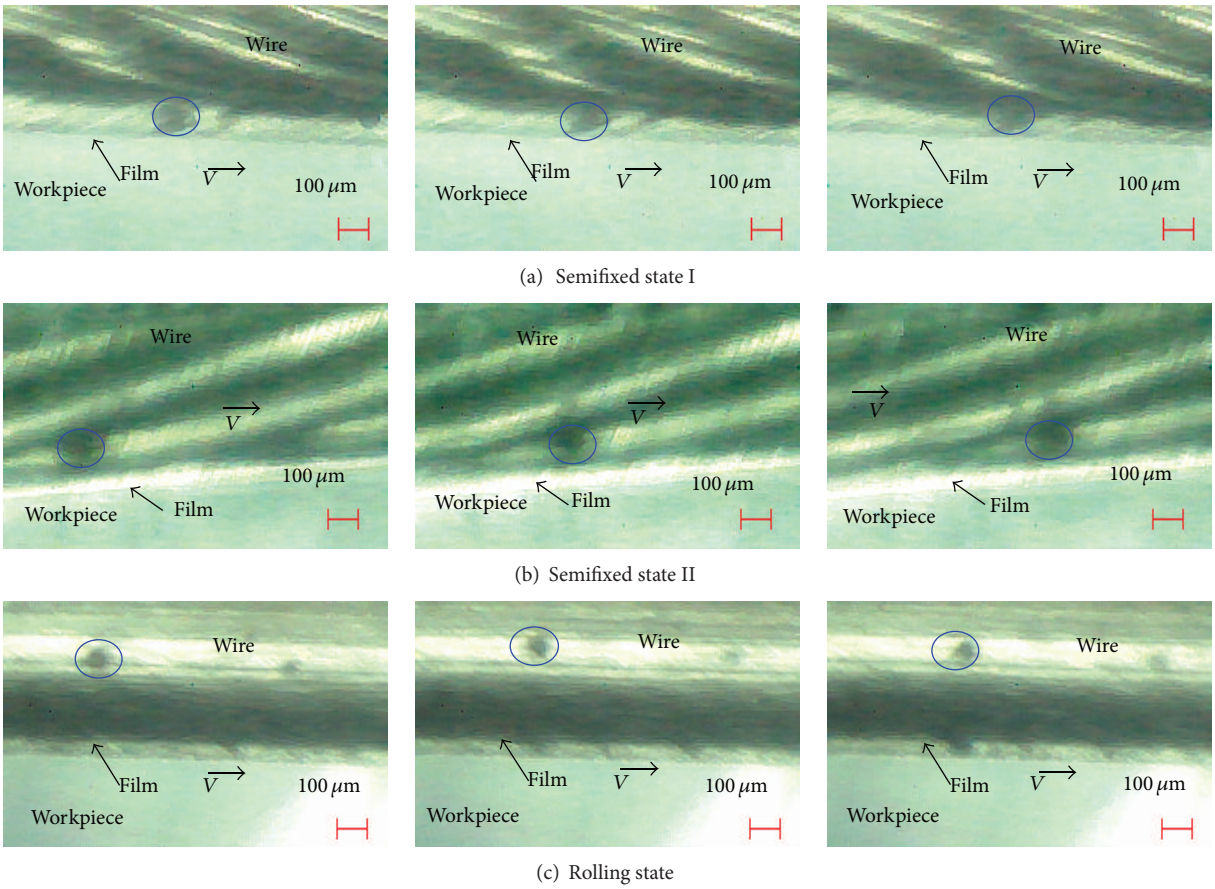


Figure 5: Comparison of the abrasive motion state between free abrasive wire saws and semifixed abrasive wire saws.

The results of general free-abrasive wire-saw studies show that the slurry film is thickest on entry and gradually diminishes in thickness throughout the cutting region. Owing to the different abrasive particle diameters, the slurry film thickness in the intermediate section is usually larger than the grain size [10] leaving some big abrasive particles aside, which results in a relatively low quantity of transient fixed abrasives. Some abrasive particles have greater thickness than the wire and, hence, are only partially in contact with the wire and the workpiece, which coincides with the results of the granulometry of the abrasives. Under the same test conditions, however, the peak value of the hydrodynamic pressure in the semifixed abrasive state is always lower than in the free-abrasive state, which has relatively small film thickness. Thus, there are more transient fixed abrasives in the former state which can be easily carried into the work region to increase the quantity because of the existing surface microgrooves. It should be noted that the transient fixed chance of wire B mentioned above is slightly higher than wire C. This is due to the fact that there is a correlation between abrasive diameter and microgroove size within the same particle-size range, and, as a result of their similar surface structures, the bigger surface microgroove size gives wire B the ability to carry more abrasive into the work area to be converted to a semifixed state. Different abrasive motion states depending on slurry hydromechanics have distinct cutting mechanisms across the processing region. Hence, the results of our study show that higher cutting efficiency and quality can be achieved by controlling the abrasive motion state in order to regulate the manner of material removal.

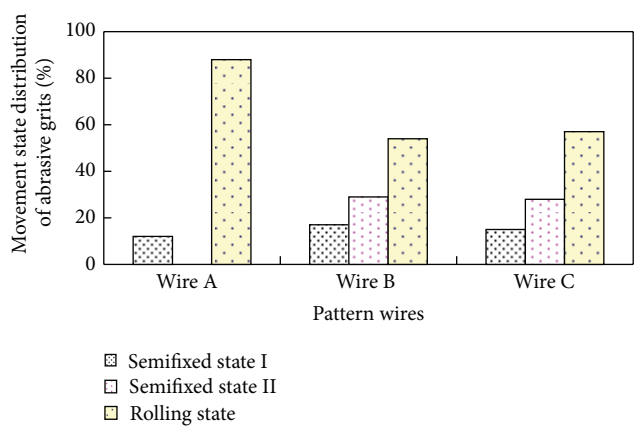


Figure 6: Statistical analysis of the abrasive motion state in the cutting process.

To study the effect of feed rate on abrasive motion state, wire B was used to simulate the cutting action on the workpiece at two different feed rates. Figure 7 illustrates the effect of sawing rate on abrasive motion state and average

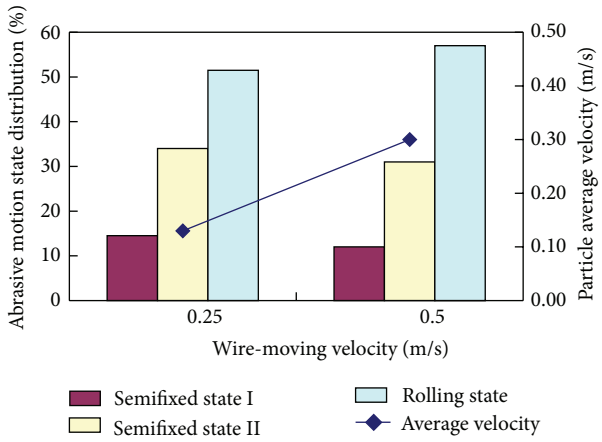


Figure 7: The effect of the wire-moving velocity on the abrasive motion state.

velocity in the mid-cutting area. The overall percentage of number of grains in the semifixed state dropped off along with the feed speed increases. Based on two typical ways in which the abrasive is fixed in the semifixed state, the average moving velocity of the abrasive is further improved along with the increase in feed rate, which is in accordance with Professor Mölle's results. The study done by Mölle assumed that, treating the abrasive suspension as a Newtonian fluid, the shear stress applied to the abrasive is proportional to the feed rate; that is, the shear stress of the abrasive particles increases with the increase in sawing rate so that the number of semifixed abrasives correspondingly decreases. This can be the important reason for the improvement in the average moving speed with the increase of feed rate.

5. Conclusion

In summary, we have investigated and proposed a novel semifixed abrasive wire-sawing technology for PV and microelectronic applications based on the above-mentioned theoretical and experimental results. This technology can increase the transient fixed abrasives by the use of the special sawing wire with a surface microgroove, which produces the "scratch" and "ploughing" action, which is akin to the fixed abrasive, to improve the slicing efficiency. In the experiment, a multistrand wire and an individual strand wire were used as the analog experimental wires. Many dynamic images, both in the semifixed and free-abrasive wire sawing processes, were captured through the dynamic three-dimensional digital microscope, which were analyzed in the computer, thereby contributing to the intuitive and profound understanding of the cutting principle of the free-abrasive wire sawing.

The main findings through the statistical analysis of the cutting process are as follows. First, there are distinct abrasive motion states, which, as mentioned above, are the semifixed state and the free-rolling state depending on the slurry hydromechanics across the work region in both the semifixed abrasive wire sawing and in the free-abrasive wire sawing. Second, the semifixed abrasive state can be classified into two categories of movement during the cutting process: the abrasives with diameters greater than the slurry film thickness between the wire and the workpiece are embedded in the microgroove, which roll with the moving wire and slide into the flank of the cutting area. Third, in comparison with the normal free-abrasive wire sawing, the semifixed abrasive wire sawing uses a different type of wire whose surface is characteristic of microgrooves texture that can make more grits turn into the transient fixed state to remove the material with a "ploughing" action. Finally, as the proportion of the semifixed state depends on the abrasive size distribution, higher cutting efficiency and quality can be achieved by controlling the abrasive diameter distribution and motion state to regulate further material removal in the semifixed process.

Acknowledgments

The work was supported by the National Natural Science Foundation of China (51075367) and Natural Science Foundation of Zhejiang Province (Y1090931).

References

[1] B. L. Lawson, N. Kota, and O. B. Ozdoganlar, "Effects of crystallography anistropy on orthogonal micromachining of single-crystal aluminum," *Transactions of the ASME, Journal of Manufacturing Science and Engineering*, vol. 130, no. 3, Article ID 031116, 11 pages, 2008.

[2] Z. J. Pei, G. R. Fisher, and J. Liu, "Grinding of silicon wafers: a review from historical perspectives," *International Journal of Machine Tools and Manufacture*, vol. 48, no. 12-13, pp. 1297–1307, 2008.

[3] S. Bhagavat and I. Kao, "A finite element analysis of temperature variation in silicon wafers during wiresaw slicing," *International Journal of Machine Tools and Manufacture*, vol. 48, no. 1, pp. 95–106, 2008.

[4] S. Bhagavat and I. Kao, "Ultra-low load multiple indentation response of materials: in purview of wiresaw slicing and other free abrasive machining (FAM) processes," *International Journal of Machine Tools and Manufacture*, vol. 47, no. 3-4, pp. 666–672, 2007.

[5] A. Bidiville, K. Wasmer, J. Michler, P. M. Nasch, M. van der Meer, and C. Ballif, "Mechanisms of wafer sawing and impact on wafer properties," *Progress in Photovoltaics: Research and Applications*, vol. 18, no. 8, pp. 563–572, 2010.

[6] H. Wu and S. N. Melkote, "Study of ductile-to-brittle transition in single grit diamond scribing of silicon: application to wire sawing of silicon wafers," *Transactions of the ASME, Journal of Engineering Materials and Technology*, vol. 134, no. 4, 2012.

[7] Z. Teng, J. He, A. J. Degnan et al., "Critical mechanical conditions around neovessels in carotid atherosclerotic plaque may promote intraplaque hemorrhage," *Atherosclerosis*, vol. 223, no. 2, pp. 321–326, 2012.

[8] H. Zhao, R. Jin, S. Wu, and J. Shi, "PDE-constrained Gaussian process model on material removal rate of wire saw slicing process," *Transactions of the ASME, Journal of Manufacturing Science and Engineering*, vol. 133, no. 2, Article ID 021012, 9 pages, 2011.

[9] J. Li, I. Kao, and V. Prasad, "Modeling stresses of contacts in wire saw slicing of polycrystalline and crystalline ingots: application to silicon wafer production," *Transactions of the ASME, Journal of Electronic Packaging*, vol. 120, no. 2, pp. 123–128, 1998.

[10] F. Yang and I. Kao, "Free abrasive machining in slicing brittle materials with wiresaw," *Transactions of the ASME, Journal of Electronic Packaging*, vol. 123, no. 3, pp. 254–259, 2001.

[11] S. Bhagavat, J. C. Liberato, C. Chung, and I. Kao, "Effects of mixed abrasive grits in slurries on free abrasive machining (FAM) processes," *International Journal of Machine Tools & Manufacture*, vol. 50, no. 9, pp. 843–847, 2010.

[12] H. J. Möller, C. Funke, M. Rinio, and S. Scholz, "Multicrystalline silicon for solar cells," *Thin Solid Films*, vol. 487, no. 1-2, pp. 179–187, 2005.

[13] P. Ge, "Some improvements on manufacturing techniques of fixed diamond wire saw," *Diamond & Abrasives Engineering*, vol. 156, no. 6, pp. 12–27, 2006.

[14] Z. Cheng, M. Yang, and R. Pei, "Hybrid machining mechanism of roll-indent-chipping in multi-wire sawing process," *Journal of Shanghai University (Natural Science)*, vol. 15, no. 5, pp. 506–511, 2009.

[15] C. Chung, C. S. Korach, and I. Kao, "Experimental study and modeling of lapping using abrasive grits with mixed sizes," *Transactions of the ASME, Journal of Manufacturing Science and Engineering*, vol. 133, no. 3, Article ID 031006, 2011.

[16] E. Teomete, "Investigation of long waviness induced by the wire saw process," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 225, no. 7, pp. 1153–1162, 2011.

[17] E. Teomete, "Roughness damage evolution due to wire saw process," *International Journal of Precision Engineering and Manufacturing*, vol. 12, no. 6, pp. 941–947, 2011.

[18] S. Y. Chen, X. Li, K. Lu, Y. Fang, and W. Wang, "Gait, stability and movement of snake-like robots," *International Journal of Advanced Robotic Systems*, vol. 9, Article ID 53627, 8 pages, 2012.

[19] C. Zhu, Q. Guan, and S. Chen, "A novel cell segmentation, tracking and dynamic analysis method in time-lapse microscopy based on cell local graph structure and motion features," in *Pattern Recognition*, vol. 321 of *Communications in Computer and Information Science*, pp. 359–366, 2012.

[20] T. Liedke and M. Kuna, "A macroscopic mechanical model of the wire sawing process," *International Journal of Machine Tools and Manufacture*, vol. 51, no. 9, pp. 711–720, 2011.