Complexity

Emerging Trends in Mechatronics

Lead Guest Editor: Aydin Azizi
Guest Editors: Gordana Collier, Abbas A. Dehghani-Sanij, Michael Modigell, Abolghasem Zabihollah, and Roya Ahmadiahangar

Emerging Trends in Mechatronics

Complexity

Emerging Trends in Mechatronics

Lead Guest Editor: Aydin Azizi
Guest Editors: Gordana Collier, Abbas A. Dehghani-Sanij, Michael Modigell, Abolghasem Zabihollah, and Roya Ahmadiahangar

Contents

Hindawi
Complexity
Volume 2021, Article ID 6617309, 12 pages
https://doi.org/10.1155/2021/6617309

Research Article

Control of Magnetic Manipulator Using Reinforcement Learning Based on Incrementally Adapted Local Linear Models

Martin Brablc,[1] **Jan Žegklitz**,[2] **Robert Grepl**,[1] **and Robert Babuška**[2,3]

[1]*Institute of Solid Mechanics, Mechatronics and Biomechanics, Faculty of Mechanical Engineering, Brno University of Technology, Brno 616 69, Czech Republic*
[2]*Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University, Prague, Prague 16 636, Czech Republic*
[3]*Cognitive Robotics, Delft University of Technology, Delft 2628 CD, Netherlands*

Correspondence should be addressed to Martin Brablc; martin.brablc@vutbr.cz

Received 11 December 2020; Revised 11 November 2021; Accepted 16 November 2021; Published 20 December 2021

Academic Editor: Aydin Azizi

Reinforcement learning (RL) agents can learn to control a nonlinear system without using a model of the system. However, having a model brings benefits, mainly in terms of a reduced number of unsuccessful trials before achieving acceptable control performance. Several modelling approaches have been used in the RL domain, such as neural networks, local linear regression, or Gaussian processes. In this article, we focus on techniques that have not been used much so far: symbolic regression (SR), based on genetic programming and local modelling. Using measured data, symbolic regression yields a nonlinear, continuous-time analytic model. We benchmark two state-of-the-art methods, SNGP (single-node genetic programming) and MGGP (multigene genetic programming), against a standard incremental local regression method called RFWR (receptive field weighted regression). We have introduced modifications to the RFWR algorithm to better suit the low-dimensional continuous-time systems we are mostly dealing with. The benchmark is a nonlinear, dynamic magnetic manipulation system. The results show that using the RL framework and a suitable approximation method, it is possible to design a stable controller of such a complex system without the necessity of any haphazard learning. While all of the approximation methods were successful, MGGP achieved the best results at the cost of higher computational complexity. Index Terms–AI-based methods, local linear regression, nonlinear systems, magnetic manipulation, model learning for control, optimal control, reinforcement learning, symbolic regression.

1. Introduction

A reinforcement learning (RL) agent interacts with the system to be controlled by measuring its states and applying actions according to a policy so that a given goal state is attained. The policy is iteratively adapted in such a way that the agent receives the highest possible cumulative reward, which is a scalar value accumulated over trajectories in the system's state space. The reward associated with each transition in the state space is described by a predefined value function.

Existing RL algorithms can be divided into critic-only, actor-only, and actor-critic variants. The critic-only variants optimize the value function (V-function) that is then used to derive the policy; the actor-only variants work directly on the policy optimization without any need for a value function; and actor-critic variants optimize both functions simultaneously. An example of the actor-only RL variant, often called Q-learning, can be found in [1] and that of the actor-critic variant in [2].

From a different point of view, RL algorithms can be also divided into model-based and model-free variants. Examples of both approaches can be found in [3, 4]. The model-based variants include a model representation of the system to be controlled and can be pretrained in simulation (offline) and then updated when controlling the actual system (online). Model-free methods learn online exclusively through trial and error. Both variants have their specific advantages and disadvantages. We can often find remarks about the model-free approach requiring much more data,

especially in high-dimensional cases [1]. In this paper, we employ the model-based, critic-only variant without any online training so that we can compare different modelling approaches.

We focus on two promising categories of approximation algorithms: genetic programming and local linear regression. Our aim is to contribute to the methodology of choosing the optimal out of dozens of existing modelling algorithms when presented with a specific RL task. This problem arises not only in connection with the RL framework (see [5]) but in modelling of a dynamical system in general [6, 7].

Genetic algorithms (GA) and their many variations are well established as a tool for modelling or parameter estimation of dynamical systems [8, 9]. However, genetic programming as a modelling approach used within RL is relatively new and promises good results with high-dimensional systems where other approaches fail. It creates a continuous-time, globally nonlinear model described by an analytical equation built of combinations of predefined functions [10]. As it is common with genetic optimization algorithms, these methods tend to be computationally demanding. On the other hand, local regression is a well-established modelling approach for model-based RF agents where the model is composed of local linear models, offering fast and computationally cheap approximation. There are several variants of local modelling methods; comprehensive examples of grid-based local linear model structure and data-based local linear regression (LLR) are described in [11, 12], respectively. Even though the use of local regression techniques within RL has been researched in the past, it was mainly based on simple, memory-based approximation methods such as the LLR, which is thoroughly described and examined in [13, 14], and more complex incremental methods such as the receptive field weighted regression (RFWR) [15, 16] or locally weighted projection regression (LWPR) [17] were omitted, with the exception of [18], where the RFWR algorithm was used as a critic approximator. The RFWR and LWPR methods provide significant benefits in lower memory use and higher stability by employing optimization-based (RFWR) or statistical (LWPR) methods to discover the optimal distribution of the local models' areas of validity, that is, the receptive fields.

It is important to benchmark the modelling methods because of the large number of existing approaches, which aim at similar tasks, while there are no simple guidelines on the method choice. Also, the presented algorithms are not yet well established within the RL domain. Finally, studying control algorithms for magnetic manipulation systems has importance on its own because of its application in many industrial fields (medical applications, magnetic levitation systems, etc.), thus leading to the two separate aims of this paper: exploring control algorithms suitable for control of precise magnetic manipulator systems and benchmarking different modelling approaches.

When dealing with real magnetic manipulator systems, we also need to address practical issues that are often neglected in simulations, that is, nonlinearities such as actuator dead zones, saturations, Coulomb friction, signal delays, and so on. These present significant obstacles then implementing the control algorithm on a real system. In some cases, the dead zone and saturation problem can be addressed by nonlinear or adaptive control laws. For example, [19] shows an approach using fuzzy control with Gaussian membership functions, which is in practice similar to the RFWR method, and [20] describes a gain-scheduling adaptive approach to deal with internal system bounds. Using RL to find a control law for a nonlinear system also has the advantage that it can often deal with such disturbances on its own through the optimization process; for example, only a limited range of the actor outputs may be limited, which is the approach utilized in this paper.

In this paper, we also present minor adjustments to the RFWR algorithm in Section 3, proposed to lower the computational complexity while preserving stability when working with low-dimensional problems.

2. Methods

2.1. Magnetic Manipulator. Genetic programming was already applied to nonlinear systems like an inverted pendulum or a collaborative robot [2, 10, 21]. To further investigate the approximation capabilities of these methods, we use a different system–a magnetic manipulator (Magman). This system consists of four coils that are independently operated by separate current controllers and a steel ball that can move freely over the coils; see Figure 1. To ensure that the ball moves only in the measured direction with limits on the edges, it is placed in a groove with 10 mm in size. In this case, we decided to limit the system to the first two coils only, as a system with four inputs is much more complex in terms of the RL computational complexity, while it does not enrich the system with different nonlinearities as it only spatially repeats the same kind of nonlinear behaviour.

The steel ball can be positioned by properly controlling the current and thereby the magnetic force of the coils. The magnetic force a coil exerts on the ball is highly dependent on the distance of the ball from the coil's centre, which introduces a significant nonmonotonic nonlinearity [22–24].

All experiments and simulation were scripted in MATLAB. The coil currents are controlled by stabilized current source modules, which communicate that MATLAB through a USB/RS232 transceiver using the virtual COM port (VCP) protocol on Windows OS. As the ball position is measured with a laser sensor with analog (voltage) output, the Humusoft MF634 IO card was used to measure the signal in real time from the MATLAB environment with a sampling period of 5 ms. Even though the Window OS is not an RTOS, with this sampling frequency, the period jitter is negligible (below 0.1%), and thus, the system can be considered real time.

Table 1 lists the parameters of the magnetic manipulator we use in our experiments. With the task being a precise positioning of an object in a magnetic field, similar concepts can be found in many real-world applications, for example, maglev, microrobots, contactless stirring of chemicals, and so on.

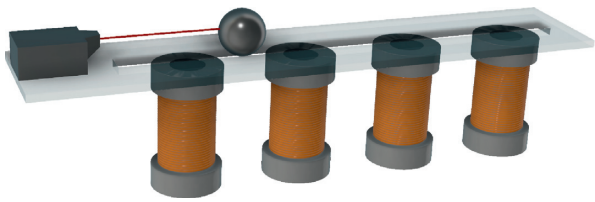


Figure 1: A schematic drawing of the magnetic manipulator system.

Table 1: Magnetic manipulator parameters.

Parameter	Value	Unit
Ball mass	53	g
Ball diameter	20	mm
Distance between edge coils and the ball position limit	20	mm
Distance between coils	25	mm
Maximal coil current	0.6	A
Sampling period	0.005	s

Approximate equations of motion inferred using the first principle method can be found in [24]. The system parameters were either measured directly or estimated using MATLAB parameter estimation toolbox based on measured data.

Generally, the system can be described by a continuous-time, nonlinear state-space model as follows:

$$\dot{x} = f(x, u), \tag{1}$$

where $\mathbf{x} = [x, \dot{x}]^\top$ is the state vector composed of the position x and velocity $\dot{x}$ of the ball, forming the continuous system state space $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^2$; $\dot{x}$ in $\dot{\mathcal{X}} \subset \mathbb{R}^2$ is the state vector derivative; and $\mathbf{u} = [u_1, u_2]^\top$ is the input (action) vector composed of the coil currents. u_1, u_2 form the system input space $\mathbf{u} \in \mathcal{U} \subset \mathbb{R}^2$. The nonlinear vector function $f: \mathcal{X} \times \mathcal{U} \longrightarrow \dot{\mathcal{X}}$ thus describes the system dynamics.

In this paper, by modelling the system, we mean approximating the underlying real function f using various methods, which all build upon experimentally measured input-output data. Each data point is formed by corresponding assumed inputs and outputs of the function f – $(\dot{\mathbf{x}}_k, \mathbf{x}_k, \mathbf{u}_k)$. In practice, these data points are corrupted by noise and other disturbances that are assumed to be with zero means.

2.2. SNGP. Single-node genetic programming (SNGP) is a graph-based genetic programming algorithm evolving a population organized as an ordered linear array of interlinked individuals, each representing a single program node [2, 10, 21]. Generally, symbolic regression algorithms try to find a model in the form of an analytical expression for a given data set by forming and evolving the expression out of elemental functions and operations. In our case, the algorithm is based on the assumption that the nonlinear function f in (1) can be efficiently approximated by the following equation:

$$\dot{x} = f(x, u) = \sum_{i,j}^{n_f, n_s} \beta_i f_i\left(x_j, u_j\right), \tag{2}$$

where the nonlinear function f_i, called the feature, is developed by means of genetic programming with n_f being the maximum number of features, n_s number of states, and the coefficients β_i estimated by the least-squares method. The features are constructed from a list of elementary functions that are assumed to be able to produce the required fitting approximation of the presented data. The features can be combined by common operators or nested, but the maximal depth of the expression is limited to avoid overfitting. The symbolic model is evolved so that the mean-squared error over the training data is minimized.

2.3. MGGP. The second GP algorithm we used is called multigene genetic programming (MGGP). As opposed to SNGP, it combines the features defined also by 2 into tree-like structured expressions called genes. The final expression is formed by a linear combination of these genes, which act as the individual features in equation (2). The parameters of this top-level linear combination are again estimated through least squares. Further details about the algorithm can be found in [25]. The actual MGGP implementation we used is extended with linear combinations of features [26] that enable the algorithm to find affine transformations of the feature space via a backpropagation-like technique, thus making it easier for the driving genetic programming algorithm to approximate the nonlinearities.

2.4. Receptive Field Weighted Regression. Receptive field weighted regression (RFWR) is an incremental approximation method that creates a set of local linear models and the corresponding Gaussian basis functions called the receptive fields and gradually updates them to fit the input-output data. The set of local linear models is updated with new data points (called the query points) using a weighted variant of the recursive least squares (RLS) method and the basis functions are updated through a gradient search with the help of heuristic decision rules. It can continually improve the set of models while still providing the best estimation of the approximated function at each query point based on the previously provided data. The original algorithm, first presented in [15, 16], which is the basis we build upon, can be best described by the following pseudocode:

(1) For each new query point $(\dot{\mathbf{x}}_k, \mathbf{x}_k, \mathbf{u}_k)$
(2) For each existing local model
(3) Calculate model weight w according to (4)
(4) If $w >$ activation limit w_{act}
(5) Update model parameters using RLS according to (6) and (7)
(6) Update the corresponding receptive field using (12) and (14)
(7) End
(8) End

(9) If no model was activated

(10) Place a new model at the query point using (15)

(11) Else if two or more models were activated with weight $w >$ pruning limit w_{prun}

(12) Prune the model with the smaller receptive field

(13) End

(14) Calculate the model output as a weighted average of the activated local models

(15) End

Usually, the receptive field activation limit is set as $w_{\text{act}} = 0.001$. This parameter represents the weight limit for a local model to be updated according to the new data and to be included in the output estimation through a weighted average with another activated model. The pruning limit is usually set as $w_{\text{prun}} = 0.7$, which represents the highest acceptable overlap of neighbouring receptive fields.

The RFWR variant described in this paper follows the main outline of the original algorithm [15] with several adjustments and improvements for the sake of stability and computational complexity for low-dimensional problems. This mainly concerns the rules for adding new local models, adjusting their receptive fields, and generalizing the algorithm in a way that the receptive fields are placed and optimized in a lower number of dimensions than the order of the models. This is especially useful in cases when the nonlinearities are significant mainly in one or two dimensions of the state space of the system. This algorithm, in its original implementation, is successfully being used to approximate inverse models of nonlinear systems to be used as a feedforward compensator [27, 28]. Figure 2 shows an example approximation of a complex univariate nonlinear function by the RFWR algorithm.

Each of the local models is represented by a parameter vector $b = [b_1, b_2, \ldots, b_n]^\top$. With the input vector (a query point) $\mathbf{X_q} = [x_1, x_2, \ldots, x_n]^\top$, the output y_q is calculated by

$$y_q = X_q^\top b. \tag{3}$$

The weight w of a local model at a query point X_q is determined by its Gaussian receptive field as follows:

$$w(X) = e^{-1/2\left(X_q - c\right)^\top C^{-1}\left(X_q - c\right)}, \tag{4}$$

with $\mathbf{c} = [c_1, c_2, \ldots, c_n]$ the vector of model centre coordinates and C^{-1} the distance inducing matrix of the basis function (receptive field). The overall output is then calculated as a weighted average of the outputs of the activated local models.

The output estimate of the set of local models and their receptive fields is calculated by the following equation:

$$\widehat{y} = \frac{1}{\sum_{i=1}^{n} w_i} \sum_{i=1}^{n} w_i y_i. \tag{5}$$

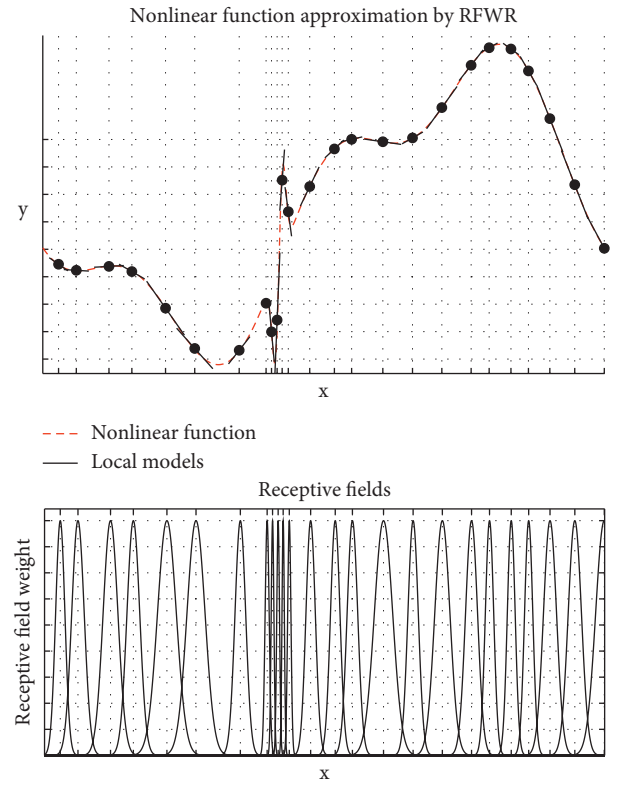


Figure 2: Example of RFWR approximation of a complex nonlinear function and the distribution of receptive fields.

We modified the original RFWR algorithm described in [15] to be used for low-dimensional problems. These modifications consist of the following:

(1) Precise placement of new receptive fields that takes into account the location and dimensions of the existing surrounding receptive fields (see section 3.3)

(2) Heuristic rules for stable updating of the receptive fields (see section 3.1)

(3) Receptive fields can be distributed along a lower number of dimensions than the dimension of the data space (see section 3.2)

2.4.1. Updating parameters of Local Models. When a new query point is acquired, the activated local models are updated using the recursive least-squares algorithm (RLS) according to

$$\begin{aligned} P^{n+1} &= \frac{1}{\lambda}\left(P^n - \frac{P^n X_q^T X_q P^n}{\lambda/w + X_q P^n X_q^T}\right), \\ b^{n+1} &= b^n + \mathrm{w}P^{n+1} X_q^T e, \\ e &= y_q - X_q b, \end{aligned} \tag{6}$$

where P is the covariance matrix of the estimate, λ is a forgetting parameter, and y_q is the acquired output for the actual system state X_q called the query point. The covariance matrix P needs is usually initialized as a diagonal matrix.

2.4.2. Updating dimensions of Basis Functions. To avoid calculating the matrix inversion in (4) for every local model, an upper triangular matrix M is used instead of C. Because of symmetry and positive definiteness, these matrices relate according to

$$C^{-1} = M^T M. \tag{7}$$

To update the receptive field, we update M using a gradient-descent optimization

$$M^{i+1} = M^i - \alpha \frac{\partial J(M)}{\partial M}, \tag{8}$$

of the cost function J as follows:

$$J = \frac{1}{\sum_{i=1}^{n} w_i} \sum_{i=1}^{n} w_i \left(y_q - y_i\right)^2, \tag{9}$$

where w_i is the activated receptive field weight, y_i is the estimated output of the respective model at the query point $(y_q; X_q)$, and n is the number of local models. The parameter α is the gradient optimization step size. As the calculation of the cost function J according to (9) is computationally very complex, we simplified the optimization algorithm through a set of heuristic decision rules and implemented the optimization as follows:

$$M^{i+1} = M^i - \alpha p \frac{\partial w(M)}{\partial M}$$

$$\frac{\partial w(M)}{\partial M} = \frac{\partial\left(e^{-1/2\left(X_q - c\right)^\top M^T M\left(X_q - c\right)}\right)}{\partial M} =$$

$$= -\left(X_q - c\right)^\top M\left(X_q - c\right) e^{-\frac{1}{2}\left(X_q - c\right)^\top M^T M\left(X_q - c\right)}. \tag{10}$$

This implementation introduces a parameter p, which is an expression of a simple heuristic to decide whether the value of a basis function (weight) at the actual query point should be increased or decreased. This enables to stop updating the distance inducing matrix when a precision criterion is met and to limit the maximal number of local models to avoid overfitting.

Parameter p can be determined by various decision rules. A simple yet effective set, which was used in this research, can be created by using a long-term (cumulated over time) MSE of a particular model according to the data points, which can be described by

$$p = \begin{cases} -1, & \text{if MSE} > \text{MSE}_{\text{lim}} \\ 1, & \text{if MSE} < \text{MSE}_{\text{lim}} \end{cases} \tag{11}$$

2.4.3. Adding New Local Models. During the optimization process, it is possible that no model exceeds the activation limit w_{act}. In such a case, a new local model with a receptive field is added to the approximation set. The centre of the receptive field is automatically placed at the actual query point, and the model parameters are initialized to fit the measured output of the approximated system. What needs to be determined is the area in the state space that should be covered by the newly created receptive field. The original algorithm uses a default diagonal distance inducing matrix for every local model. However, an optimal distance inducing matrix can be determined. Intuitively, the new receptive field should cover the gap between the already existing models. The distance inducing matrix should be initialized as a diagonal matrix with parameters that ensure that the new receptive field does not overlap with any existing one more than a preset limit. In our case, the limit was set to $0.5w_{\text{prun}}$. Since this would be a complex optimization task not suitable for real-time calculation, we simplified the criterion so that the maximal overlapping weight of two models is analyzed only over the line segment connecting their centres. In that case, the distance parameter for initializing the distance inducing matrix can be determined by the following equation.where $v_i = c_n - c_i$ is a vector between the new centre c_n and the centre of a neighbouring receptive field c_i. A two-dimensional example is shown in Figure 3. This method yields a better estimate of the distance inducing matrix of the new receptive field than the fixed initial dimension matrix in the original algorithm as it requires fewer iterations to stabilize and to cover the gap between neighbouring receptive fields.

$$d_i = -\frac{2\,\log\left(w_{\text{prun}}/2\right)}{\left|v_i\right| - \sqrt{-2\,\log\left(w_{\text{prun}}/2\right)/v_i M_i^T M_i v_i}}, \tag{12}$$

The distance parameter d_i has to be calculated for every existing local model, and the minimal distance d_{min} is used to initialize the distance inducing matrix according to

$$M_n^0 = I\sqrt{d_{\text{min}}}, \tag{13}$$

where I is a unity matrix of the corresponding order.

In the specific case of the magnetic manipulator, the inputs of the local models would correspond to (x, u) and the output to $\dot{x}$.

2.5. Reinforcement Learning. Consider the following discrete deterministic state-space model of a system to be controlled:

$$x_{k+1} = f\left(x_k, u_k\right), \tag{14}$$

where $k \in \mathbb{Z}$ denotes discrete time instants, $x_k, x_{k+1} \in \mathcal{X} \subset \mathbb{R}^n$ is the state vector, and $u_k \in \mathcal{U} \subset \mathbb{R}^m$ is the input vector. An RL agent learns to control the system so that it achieves the maximal cumulated reward on a trajectory from the initial state to the desired state [10]. At each state transition, as described by (14), the agent receives a scalar reward according to

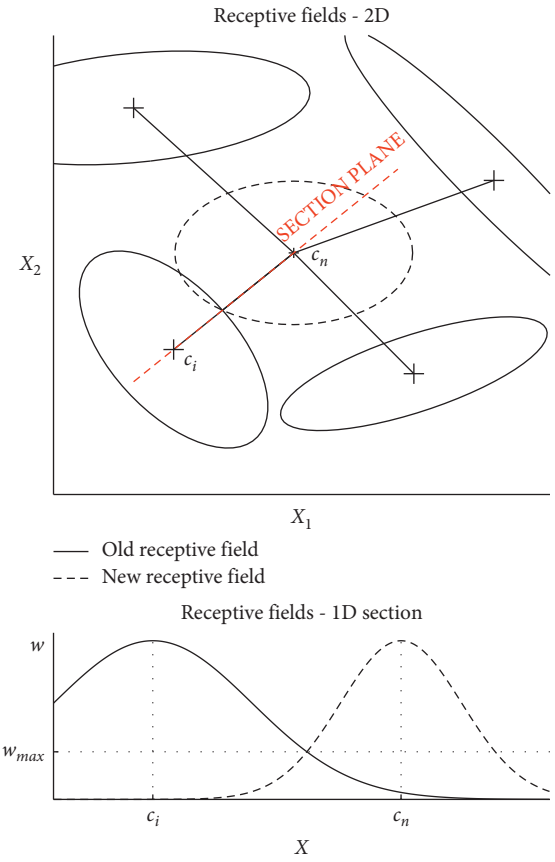


FIGURE 3: Demonstration of adding a new reference field to the set of RFWR local models and matching its dimensions to fit the surrounding reference fields.

$$r = \rho(x_k, u_k, x_{k-1}). \tag{15}$$

The reward function ρ is usually based on the distance of the current state to the goal state. The optimal control law, called the policy, π: $\mathcal{X} \longrightarrow \mathcal{U}$ is determined as follows such that it maximizes the cumulative reward, called the return:

$$R^{\pi} = E\left\{\sum_{k=0}^{\infty} \gamma^k \rho(x_k, \pi(x_k), x_{k+1})\right\}, \tag{16}$$

where $\gamma \in (0, 1)$ is the discount factor and the initial state x_0 is selected from the state space domain $\mathcal{X}$. The return for any permissible initial state x is captured by the value function V: $\mathcal{X} \longrightarrow \mathbb{R}$ defined as follows:

$$V(x) = E\left\{\sum_{k=0}^{\infty} \gamma^k \rho(x_k, \pi(x_k), x_{k+1})\right\}, \quad x_0 = x. \tag{17}$$

An approximation of the optimal V-function $\widehat{V}(x)$ can be found by solving the Bellman equation as follows:

$$\widehat{V}(x) = \max_{u \in \mathcal{U}} [\rho(x, \pi(x), f(x, u)) + \gamma \widehat{V}(f(x, u))]. \tag{18}$$

The optimal action can be found as the action that steers the system to a state with maximal value [21]. This corresponds to maximization of the right-hand side of (18):

$$u = \arg\max_{u' \in U} [\rho(x, u', f(x, u')) + \gamma V(f(x, u'))]. \tag{19}$$

3. Experimental Results

We prepared training and validation I/O data sets measured on the magnetic manipulator with random input signals as a list of data points in the form $(\dot{x}_k, x_k, u_k)$. The random input signals (coil currents) were generated in the way that only one coil was active at a time that eliminated possible electromagnetic interactions between them (the coil current was controlled by an HW-based current feedback controller module rendering the transient times negligible). Figure 4 shows an example of a training data set.

Even though the ball's position measurement is very precise, it still contains significant noise. For that reason, the time-domain derivatives of the position (velocity and acceleration) needed for the dynamic model approximation were determined using the Savitzky–Golay filter, which is an FIR filter based on least-squares polynomial approximation able to perform numerical differentiation while filtering the noise simultaneously [29, 30].

Especially for the RFWR implementation, it is important to note that the system's nonlinearity is mainly significant along the position of the ball and the system can be seen as linear in parameters along the other dimensions (acceleration and velocity). In this case, the general model (1) can be rearranged as follows:

$$\ddot{x} = f(x, u) + b_0 \operatorname{sign}(\dot{x}) + b_1 \dot{x} + b_2 \dot{x}^2, \tag{20}$$

where the function $f(x, u)$ represents the significant nonlinearity suitable for local approximation, the term $b_0 \operatorname{sign}(\dot{x})$ represents a simple model of dry friction, the term $b_1 \dot{x}$ represents the viscous friction, and the last term $b_2 \dot{x}^2$ models nonlinear damping caused by electromagnetic induction influencing the steel ball while moving rapidly through a magnetic field. Despite being nonlinear, all of the terms are linear in their parameters and can be modelled globally, which means that the local models share parameters b_0 through b_2.

The term $b_0 \operatorname{sign}(\dot{x})$ in the (20) is quite important in practical situations where Coulomb friction is not negligible. The sign function is often being used to approximate the effects of Coulomb friction wherever there is no significant stiction (difference between static and dynamic friction effects). There are better approximations for simulation purposes, for example, the sigmoid function; however, most of them are not linear in parameters and thus not applicable for RLS parameter estimation.

The same data set was presented to all of the approximation methods (RFWR, SNGP, and MGGP). Due to the stochastic nature of the two algorithms based on genetic programming, the same process was repeated with different pseudorandom seeds. Overall, 30 runs for SNGP and MGGP and 1 run for RFWR were made. Table 2 shows the summary of the MSE results.

Since the MSE of the models with respect to the training set is not sufficient to decide which models are better, two

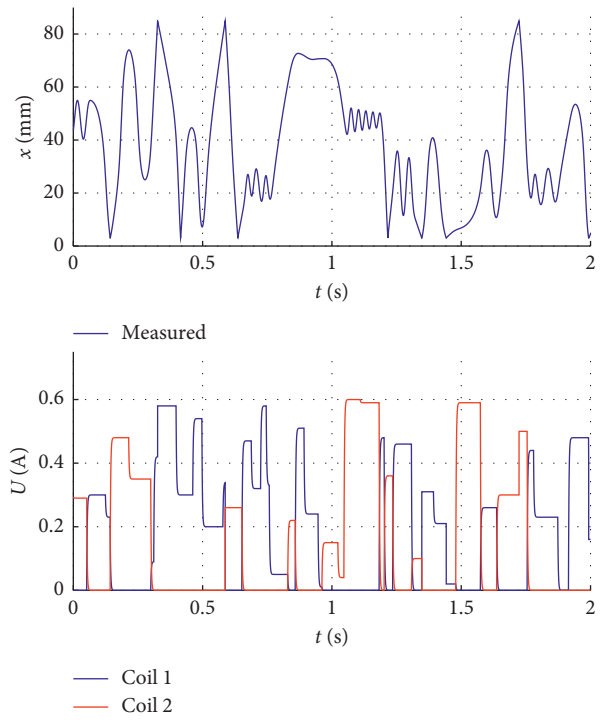


Figure 4: Random input signal (generalized binary noise with random coil switching) with only one coil active at a time for training and validation sets generation.

Table 2: Statistical results of the best models.

Model	Training MSE	5-SAP validation MSE	Control MSE
SNGP 3	$6.53 \cdot 10^{-2}$	$5.71 \cdot 10^{-12}$	$4.02 \cdot 10^{-5}$
SNGP 7	$6.78 \cdot 10^{-2}$	$5.77 \cdot 10^{-12}$	$2.46 \cdot 10^{-4}$
SNGP 10	$5.89 \cdot 10^{-2}$	$5.36 \cdot 10^{-12}$	$4.23 \cdot 10^{-5}$
SNGP 13	$7.28 \cdot 10^{-2}$	$6.37 \cdot 10^{-12}$	$4.28 \cdot 10^{-5}$
SNGP 17	$7.48 \cdot 10^{-2}$	$5.84 \cdot 10^{-12}$	$1.28 \cdot 10^{-4}$
SNGP 18	$7.23 \cdot 10^{-2}$	$5.79 \cdot 10^{-12}$	$2.06 \cdot 10^{-4}$
SNGP 21	$7.65 \cdot 10^{-2}$	$6.41 \cdot 10^{-12}$	$3.40 \cdot 10^{-3}$
SNGP 27	$5.32 \cdot 10^{-2}$	$6.57 \cdot 10^{-12}$	$1.16 \cdot 10^{-4}$
SNGP 29	$5.85 \cdot 10^{-2}$	$6.19 \cdot 10^{-12}$	$9.60 \cdot 10^{-5}$
SNGP 30	$6.93 \cdot 10^{-2}$	$5.99 \cdot 10^{-12}$	$3.75 \cdot 10^{-5}$
MGGP 5	$5.94 \cdot 10^{-2}$	$5.49 \cdot 10^{-12}$	$6.34 \cdot 10^{-5}$
MGGP 8	$4.87 \cdot 10^{-2}$	$5.13 \cdot 10^{-12}$	$4.14 \cdot 10^{-5}$
MGGP 9	$5.69 \cdot 10^{-2}$	$6.18 \cdot 10^{-12}$	$3.80 \cdot 10^{-5}$
MGGP 13	$6.59 \cdot 10^{-2}$	$5.67 \cdot 10^{-12}$	$4.13 \cdot 10^{-5}$
MGGP 16	$5.96 \cdot 10^{-2}$	$5.29 \cdot 10^{-12}$	$6.59 \cdot 10^{-5}$
MGGP 17	$5.99 \cdot 10^{-2}$	$5.08 \cdot 10^{-12}$	$5.46 \cdot 10^{-5}$
MGGP 18	$7.35 \cdot 10^{-2}$	$6.28 \cdot 10^{-12}$	$1.50 \cdot 10^{-4}$
MGGP 21	$5.60 \cdot 10^{-2}$	$5.92 \cdot 10^{-12}$	$6.74 \cdot 10^{-5}$
MGGP 22	$5.55 \cdot 10^{-2}$	$5.74 \cdot 10^{-12}$	$8.84 \cdot 10^{-5}$
MGGP 23	$6.00 \cdot 10^{-2}$	$5.82 \cdot 10^{-12}$	$3.56 \cdot 10^{-5}$
RFWR	$6.25 \cdot 10^{-2}$	$1.24 \cdot 10^{-11}$	$6.57 \cdot 10^{-5}$

separate data sets were measured: one was used for the training of the models and one for validation. However, since the magnetic manipulator is not open-loop stable, the common open-loop validation is not suitable, as every model diverges quickly even though the parameters may be close to ideal due to errors introduced by numerical integration. Therefore, the models were validated in several steps-ahead prediction mode. As we also suspected that only one-sample-ahead validation could be influenced by the remaining noise in the measured signals, we validated the model for 1, 3, 5, 10, 50, 100, and 250 samples ahead. We used the five-step-ahead prediction (5-SAP) as a baseline for selecting the best models for further experiments. The reason to choose five samples is based on an experimentally

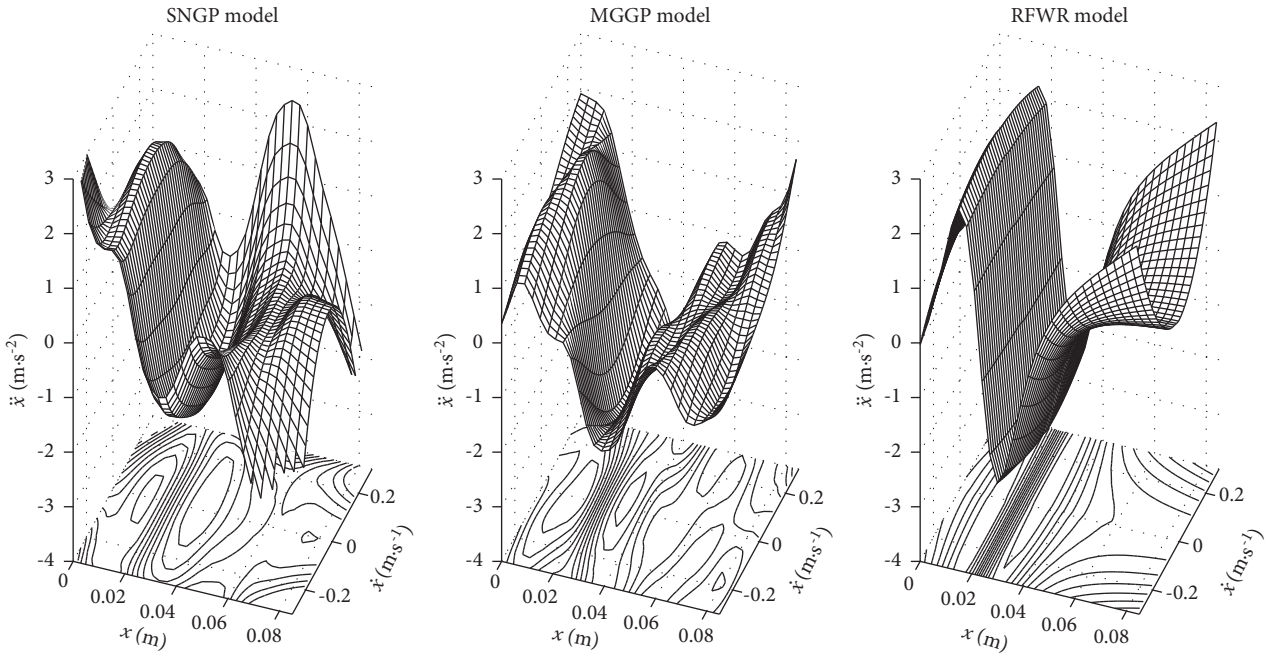


Figure 5: Comparison of approximated dynamic models for the SNGP, MGGP, and RFWR method, best-fitting models from each category. The 3D visualization is shown for constant input $u = [0, 0.6]^T$.

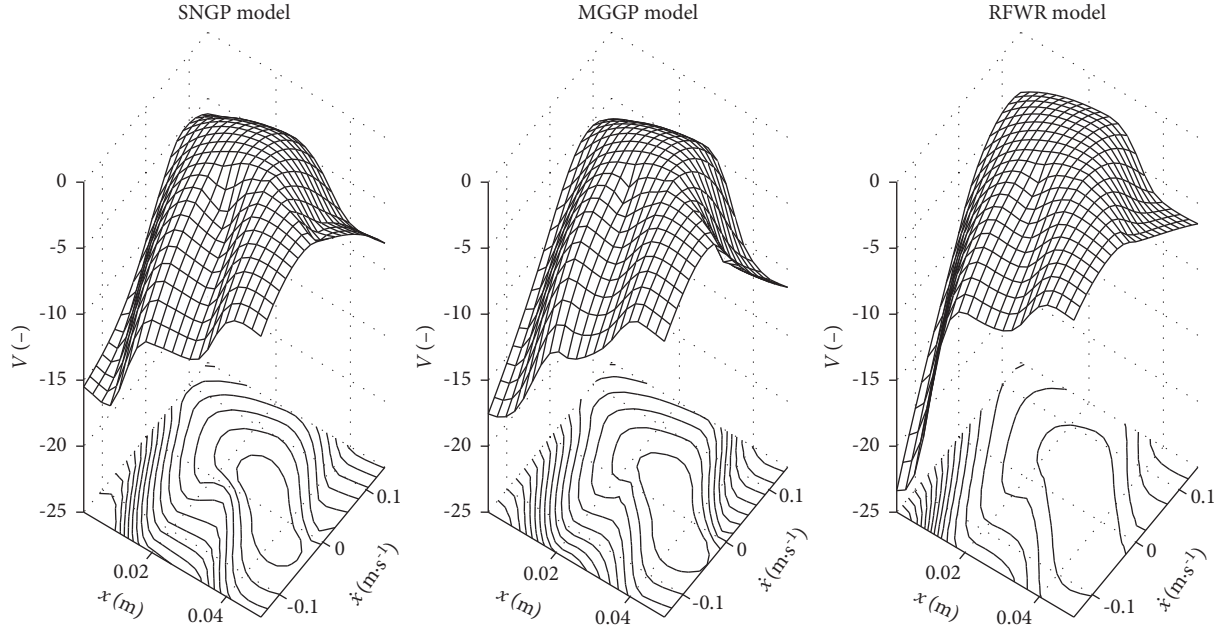


Figure 6: Comparison of V-functions based on the best fitting models from each of the approximation methods: SNGP, MGGP, and RFWR.

validated assumption that shorter intervals do not show the model's imprecision and longer intervals cause even very precise models to diverge randomly.

The n-step-ahead prediction validation is based on a moving frame of n consecutive data points, where the first data point is applied as the initial condition for numerical integration (using the ode45 solver) of the dynamical model being tested. When the simulation reaches the n-th step, an MSE residual is calculated between the corresponding data point and the model prediction. The resulting model validation metric is then calculated as the sum of residuals over each of the prediction frames.

Figure 5 shows examples of the models constructed by each algorithm. As the transition model of the system is four-dimensional, for visualization purposes, the figures show a two-dimensional situation for the input vector set to $u = [0, 0.6]^T$. This corresponds to the situation when the first coil is turned off and the current through the second coil is

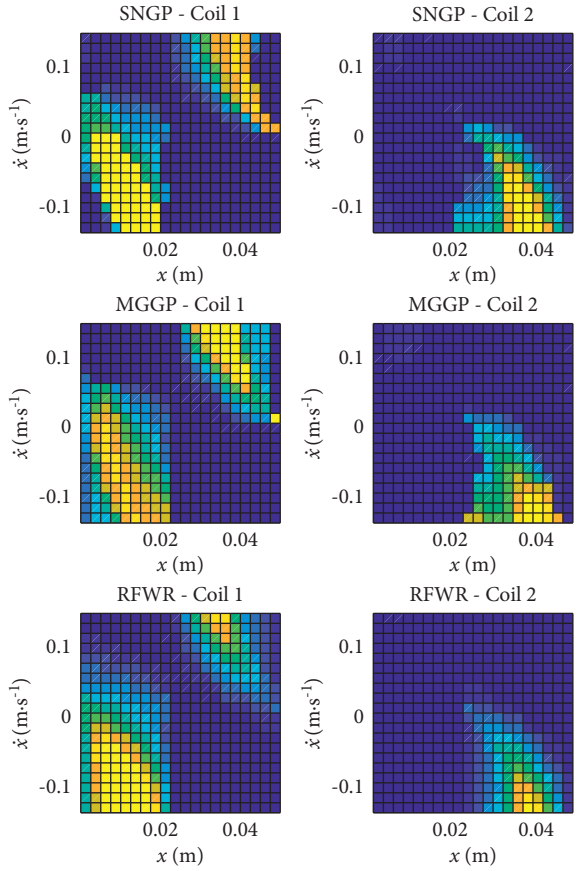


Figure 7: Comparison of policies based on the best fitting models of the magnetic manipulator from each of the approximation methods.

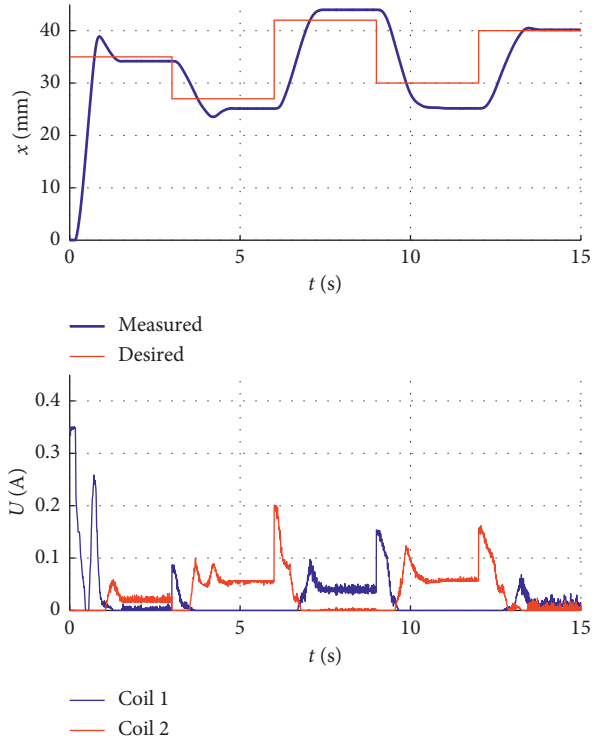


Figure 8: Example of the control performance for a controller based on the SNGP 10 model.

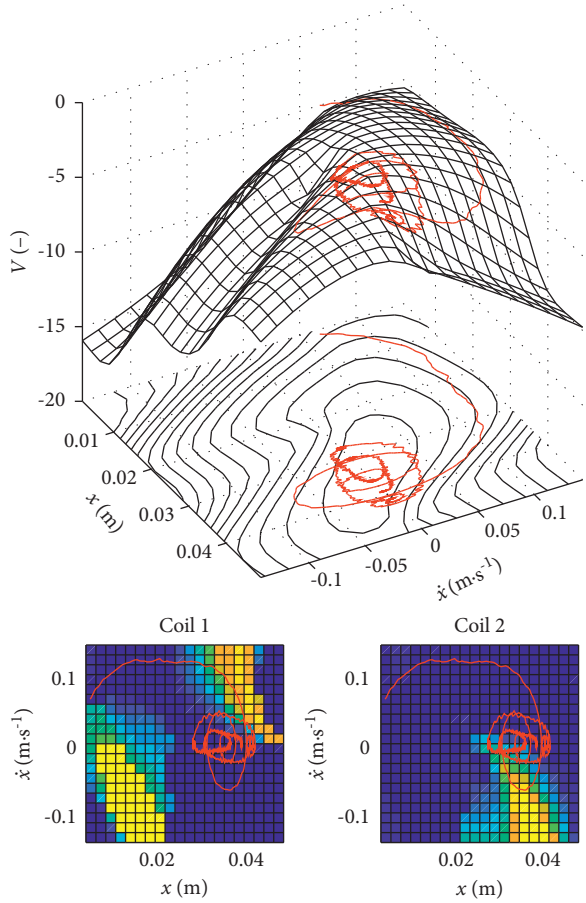


Figure 9: Example of the control sequence trajectory plotted over the policy and the value function based on the model SNGP 10.

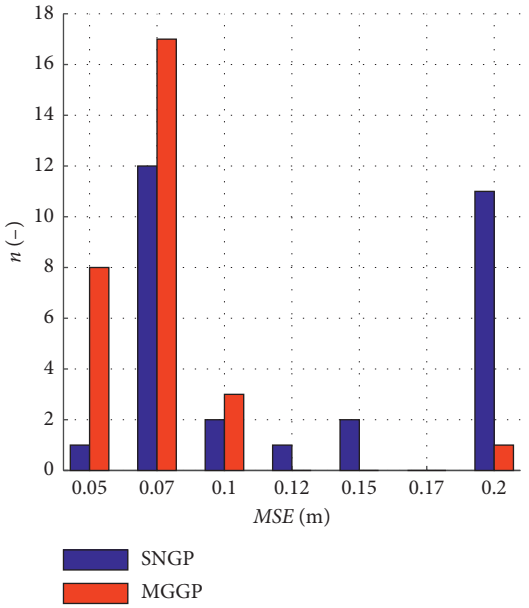


Figure 10: Histogram dividing the tested models into several categories according to their MSE.

set to the maximal value. The plotted functions were cropped in the parts of the state space that are not reachable by the system (typically high velocities close to the edge of the frame).

Based on the 5-SAP validation results, 21 best models (10x SNGP, 10x MGGP, and 1x RFWR) were chosen to be used for RL control of the magnetic manipulator. Based on these models, we calculated the approximations of the optimal V-functions using the fuzzy value iteration [10, 21]. Furthermore, we used equation (19) to calculate the corresponding policies. Figure 6 shows examples of the value functions that resulted in the policies shown in Figure 7.

All the policies were tested on the actual magnetic manipulator. A sequence of five consecutive goal states was chosen as a goal state trajectory, and a corresponding V-function and policy were calculated for each one. During the actual control process, a new policy is used every time the goal state changes. Figure 8 shows an example of the control experiments with an RL controller based on the model SNGP 10. Furthermore, Figure 9 shows the ball's trajectory in the state space plotted over the V-function and the policy.

To conduct all of the experiments, we measured the ball's position using a laser distance sensor, and a PCIe I/O card Humusoft MF634 was used to capture the sensor's analog output signal. The coil currents were driven by a custom dual-channel current controller. Both of the devices are operated from MATLAB.

3.1. Results. We compared the performance of various models of a complex nonlinear system created with three different approximation methods, two of which were based on genetic programming and the third was based on a modified local linear approximation algorithm (RFWR). Based on these models, we used a model-based critic-only RL agent to control the system and validate the results.

Table 2 shows the resulting MSE values from the estimation, validation, and control processes.

Most of the models selected for the actual control experiments were successful in achieving stable control, although they differ in precision. The histogram in Figure 10 shows the number of models for the two GP-based algorithms (SNGP and MGGP), which fall into several MSE categories. The MSE describes the control precision as the mean-squared error between the goal and the actual trajectory of the closed-loop controlled system.

4. Conclusions

First of all, the results show that it is possible to construct even such a complex nonlinear system using the RL framework. The results are not significant in terms of control precision, which depends highly on the specific system, amount of experimental data, and many other factors. An important achievement is a fact that all of the modelling algorithms demonstrated in this paper provide a viable alternative to the commonly used methods while being much less computationally expensive (in the case of RFWR) or much more user-friendly (in the cases of SNGP and MGGP). Also, it was proven that the commonly used RL framework may be built even on tom of imperfect models.

It is clear from the results that the methods that at least partially depend on random number generation (SNGP and MGGP) need to be run repeatedly in search for the best solution, which clearly outperforms the result of the local approximation method (RFWR). On the other hand, the RFWR method requires significantly less computational power than the GP-based methods. Also, it seems that both the SNGP and the MGGP are able to find similarly precise models with the MGGP having a higher probability of converging to the best solution. It is interesting to note that models with better training or validation fit are not always better for control, as can be seen in Table 2. All of the methods provide a useful tool to be used within the reinforcement learning framework with the main advantage of the GP-based approximation method being a form of output (analytical expression) that is understandable and readable and whose complexity is controllable through intuitive parameters. Considering the simplifications made during the simulations and experiments and a relative imprecision during the control processes, there is space for future research in modifying these methods to be suitable for higher-dimensional systems, implementing GPU, handling specific nonlinearities (friction, hysteresis, etc.), and using them also for an approximation of the V-functions and policies.

Data Availability

Research data in the form of simulation and experimental results and MATLAB files are available from the corresponding (first) author upon request (martin.brablc@vutbr.cz).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] S. Gu, T. Lillicrap, I. Sutskever, and S. Levine, "Continuous deep Q-learning with model-based acceleration," in *Proceedings of the Proc. 33nd Int. Conf. Mach. Learn. New York, USA: JMLR.org*, pp. 2829—–2838, ACM, New York NY USA, 19 June 2016, https://arxiv.org/abs/1603.00748.

[2] E. Alibekov, J. Kubalik, and R. Babuska, "Symbolic method for deriving policy in reinforcement learning," in *Proceedings of the 2016 IEEE 55th Conf. Decis. Control*, pp. 2789–2795, IEEE, Las Vegas, NV, USA, 12 December 2016, https://ieeexplore.ieee.org/document/7798684/.

[3] I. Koryakovskiy, M. Kudruss, R. Babuška et al., "Benchmarking model-free and model-based optimal control," *Robotics and Autonomous Systems*, vol. 92, pp. 81–90, 2017, https://linkinghub.elsevier.com/retrieve/pii/S0921889016301592.

[4] M. P. Deisenroth and C. E. Rasmussen, "PILCO: a model-based and data-efficient approach to policy search," in *Proceedings of the Proc. 28th Int. Conf. Int. Conf. Mach. Learn*, pp. 465—–472, Omnipress, Bellevue Washington USA, 28 June 2011, https://dl.acm.org/citation.cfm?id=3104482.3104541.

[5] Y. Duan, X. Chen, R. Houthooft, J. Schulman, and P. Abbeel, "Benchmarking deep reinforcement learning for continuous control," in *Proceedings of the Proc. 33rd Int. Conf. Int. Conf. Mach. Learn*, vol. 48, pp. 1329–1338, JMLR.org, New York, USA, 19 June 2016, https://dl.acm.org/citation.cfm?id=3045390.3045531http://arxiv.org/abs/1604.06778.

[6] J. Peters and S. Schaal, "Policy gradient methods for Robotics," in *Proceedings of the 2006 IEEE/RSJ Int. Conf. Intell. Robot. Syst*, pp. 2219–2225, IEEE, Beijing, China, 9 October 2006, https://ieeexplore.ieee.org/document/4058714/.

[7] A. Azizi, F. Entessari, K. G. Osgouie, and A. R. Rashnoodi, "Introducing neural networks as a computational intelligent technique," *Applied Mechanics and Materials*, vol. 464, p. 11, 2013.

[8] A. Azizi, "Applications of artificial intelligence techniques to enhance sustainability of industry 4.0: design of an artificial neural network model as dynamic behavior optimizer of robotic arms," *Complexity*, vol. 2020, p. 3, 2020.

[9] *Applications of Artificial Intelligence Techniques in Industry 4.0*, Springer, Singapore, 2019.

[10] J. Kubalík, E. Alibekov, and R. Babuška, "Optimal control via reinforcement learning with symbolic policy approximation," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4162–4167, 2017, https://linkinghub.elsevier.com/retrieve/pii/S2405896317312594.

[11] I. Grondman, M. Vaandrager, L. Busoniu, R. Babuska, and E. Schuitema, "Efficient model learning methods for actor-critic control," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 3, pp. 591–602, 2012, https://ieeexplore.ieee.org/document/6096441/.

[12] I. Grondman, L. Busoniu, and R. Babuska, "Model learning actor-critic algorithms: performance evaluation in a motion control task," in *Proceedings of the 2012 IEEE 51st IEEE Conf. Decis. Control*, pp. 5272–5277, IEEE, Maui, HI, USA, 10 December 2012, https://ieeexplore.ieee.org/document/6426427/.

[13] C. G. Atkeson, A. W. Moore, and S. Schaal, "Locally weighted learning for control," *Lazy Learning*, vol. 11, no. 1-5, pp. 75–113, 1997, https://doi-org.ezproxy.lib.vutbr.cz/10.1023/A:1006511328852.

[14] L. Buşoniu, B. De Schutter, and R. Babuška, *Approximate Dynamic Programming and Reinforcement Learning*, pp. 3–44, Springer, Berlin, Heidelberg, 2010, https://link.springer.com/10.1007/978-3-642-11688-9_1.

[15] S. Schaal and C. G. Atkeson, "Constructive incremental learning from only local information," *Neural Computation*, vol. 10, no. 8, pp. 2047–2084, nov 1998, https://www.mitpressjournals.org/doi/10.1162/089976698300016963.

[16] J. Su, J. Wang, and Y. Xi, "Incremental learning with balanced update on receptive fields for multi-sensor data fusion," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 1, pp. 659–665, feb 2004, https://ieeexplore.ieee.org/document/1262536/.

[17] S. Vijayakumar, A. D'souza, T. Shibata, J. Conradt, and S. Schaal, "Statistical learning for humanoid robots," *Autonomous Robots*, vol. 12, no. 1, pp. 55–69, 2002, https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4359172.

[18] D.-H. Lee and J.-J. Lee, "Incremental receptive field weighted actor-critic," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 62–71, feb 2013, https://ieeexplore.ieee.org/document/6246695/.

[19] M. Latifinavid, A. Donder, and E. ilhan Konukseven, "High-performance parallel hexapod-robotic light abrasive grinding using real-time tool deflection compensation and constant resultant force control," *International Journal of Advanced Manufacturing Technology*, vol. 96, p. 6, 2018.

[20] T. Yang, N. Sun, Y. Fang, X. Xin, and H. Chen, "New adaptive control methods for *n*-link robot manipulators with online gravity compensation: design and experiments," *IEEE Transactions on Industrial Electronics*, vol. 69, pp. 1–2022.

[21] E. Alibekov, J. Kubalík, and R. Babuška, "Policy derivation methods for critic-only reinforcement learning in continuous spaces," *Engineering Applications of Artificial Intelligence*, vol. 69, pp. 178–187, 2018, https://www.sciencedirect.com/science/article/pii/S0952197617302993?via%3Dihub.

[22] Z. Hurak and J. Zemanek, "Feedback linearization approach to distributed feedback manipulation," in *Proceedings of the 2012 Am. Control Conf.*, pp. 991–996, IEEE, Montreal, QC, Canada, 27 June 2012, https://ieeexplore.ieee.org/document/6315262/.

[23] J. Zemánek, S. Čelikovský, and Z. Hurák, "Time-optimal control for bilinear nonnegative-in-control systems: application to magnetic manipulation," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 16032–16039, 2017, https://linkinghub.elsevier.com/retrieve/pii/S2405896317325430.

[24] J.-W. Damsteeg, S. P. Nageshrao, and R. Babuska, "Model-based real-time control of a magnetic manipulator system," in *Proceedings of the 2017 IEEE 56th Annu. Conf. Decis. Control*, pp. 3277–3282, IEEE, Melbourne, VIC, Australia, 12 December 2017, https://ieeexplore.ieee.org/document/8264140/.

[25] M. Hinchliffe, H. Hiden, B. McKay, M. Willis, M. Tham, and G. Barton, "Modelling chemical process systems using a multi-gene genetic programming algorithm – BibSonomy," in *Proceedings of the Late Break. Pap. Genet. Program. 1996 Conf.*, J. R. Koza, Ed., pp. 56—–65, Stanford Bookstore, Stamford, CA, USA, 28 July 1996, https://www.bibsonomy.org/bibtex/2ba500b4ed22826a3b171019d4a172229/brazovayeye.

[26] J. Žegklitz and P. Pošík, "Linear combinations of features as leaf nodes in symbolic regression," in *Proceedings of the Proc. Genet. Evol. Comput. Conf. Companion - GECCO '17*, pp. 145-146, ACM Press, New York, USA, 15 July 2017, https://dl.acm.org/citation.cfm?doid=3067695.3076009.

[27] R. Grepl, V. Sova, and J. Chalupa, "Adaptive control of electro-mechanical actuator using receptive field weighted regression," *Advanced Mechatronics Solutions*, vol. 393, pp. 621–626, 2016, https://link.springer.com/10.1007/978-3-319-23923-1_87.

[28] R. Grepl and B. Lee, "Modeling, parameter estimation and nonlinear control of automotive electronic throttle using a Rapid-Control Prototyping technique," *International Journal of Automotive Technology*, vol. 11, no. 4, pp. 601–610, aug 2010, https://link.springer.com/10.1007/s12239-010-0072-7.

[29] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, jul 1964, https://pubs.acs.org/doi/abs/10.1021/ac60214a047.

[30] M. Brablc, V. Sova, and R. Grepl, "Adaptive feedforward controller for a DC motor drive based on inverse dynamic model with recursive least squares parameter estimation," in *Proceedings of the Proc. 2016 17th Int. Conf. Mechatronics - Mechatronika, ME*, D. Maga and T. Brezina, Eds., pp. 146–150, IEEE, Prague, Czech Republic, 7 December 2016, https://www.scopus.com/inward/record.url?eid=2-s2.0-85015277645&partnerID=MN8TOARS.

Hindawi
Complexity
Volume 2021, Article ID 6670625, 10 pages
https://doi.org/10.1155/2021/6670625

Research Article

Cyber-Security Assessment of Industry 4.0 Enabled Mechatronic System

Piotr Kotuszewski, Krzysztof Kukielka, Paweł Kluk, Andrzej Ordys, Karol Bieńkowski, Jan Maciej Kościelny, Michał Syfert, Paweł Wnuk, Jakub Możaryn, and Bartłomiej Fajdek

Faculty of Mechatronics, Warsaw University of Technology, ul. Sw Andrzeja Boboli 8, Warszawa 02-225, Poland

Correspondence should be addressed to Andrzej Ordys; andrzej.ordys@pw.edu.pl

Received 11 December 2020; Revised 30 August 2021; Accepted 1 October 2021; Published 22 October 2021

Academic Editor: Aydin Azizi

This paper reports on development of a laboratory set-up, for testing the concept and the components of Industry 4.0 enabled mechatronic system. A simple mock manufacturing lane is connected to a collaborating robot. There are three different programmable logic controllers (PLC) and various other elements which need to talk to each other using a variety of communication protocols. The controllers, and consequently the schedule of operations, can be programmed remotely. Moreover, the information is exchanged through the cloud service which could also assume the role of a soft-PLC. From the perspective of cyber-security, this set-up enables defining benchmarks and performing tests for resilience of the control structure. Some test scenarios are discussed in the paper.

1. Introduction

1.1. Industry 4.0. In the modern, fast-growing world, enterprises face many challenges such as dealing with large amounts of data, speed of making the right decisions, or flexibility of production processes [1]. The main reason for this is the highly competitive production environment. Very important is the concept of production flexibility, which means abandoning mass production and, instead, targeting individual products to customer's request [2]. Such a change results in a shorter product life cycle and increases the overall product range. However, it is required to change technological devices and machines to more flexible ones, adapted to different operating modes. Robotic and more broadly mechatronic devices play a pivotal role in this process. In addition, innovative tools and platforms should be provided for mutual cooperation of all areas of business operations [1]. To understand the needs of and subsequent creation of new and modification of existing manufacturing processes, a new concept of industry, i.e., Industry 4.0, describing the changes in the manufacturing, leading to so-called 4th industrial revolution, has emerged [3, 4].

As a result of this latest industrial revolution, factories are to be created where intelligent networks connect processes, machines, products, suppliers, and customers [2]. The production lines of the future can be supervised and managed remotely, and the final product is adapted to the needs of a given client on an ongoing basis [5].

1.2. Cyber-Physical Systems. Cyber-physical systems have been identified as one of the key research areas by the European Union Research programmes as well as by National Science Foundation, USA. This is a vibrant and relevant field of research which is likely to dominate in control systems design for the years to come, as it stems from rapid progress in communication, computing, and networking. Technologies underpinning the cyber-physical systems include sensors and wireless sensor networks, communication protocols, distributed control systems, and cloud computing.

Such systems are one of the pillars of Industry 4.0 manufacturing processes. Current trends in manufacturing systems include Internet of Things (IoT), cloud computing, mobile devices, and big data [2].

There are many examples of application areas, for instance, automotive systems-optimization of power train of hybrid vehicles [6], collaborating robotic systems for smart production [7], robotic systems for medical applications [8], distributed power generation (e.g., wind turbines) [9], and smart homes [10], to mention just a few.

1.3. Industrial Internet of Things (IIoT). One of the paradigms of Industry 4.0 is the development and implementation of Industrial Internet of Things. This modification of the Internet of Things technology is intended for industrial applications.

The basic idea of the Internet of Things is to collect data. In the case of the Industrial Internet of Things, this is the collection of large amounts of process data and their transmission to data centers [1].

Data from various sources are collected and analyzed in the cloud. Data sources include devices such as sensors, actuators, PLCs, industrial robots, production equipment (e.g., CNC milling machines), and mobile robots [11].

The Industrial Internet of Things is just taking the shape of useful technology in industry. To analyze the data, the IIoT device connects to the cloud [11]. This is a big challenge, because it is necessary to integrate the new technology into the existing infrastructure and, additionally, ensure data security.

A similar trend, although not always using the name of cyber-physical systems, has been observed in industrial control systems. This trend is to control installations remotely, using sensors and actuators connected with the controller via a wireless network. The rationale is to reduce the costs of workforce and, also, to remove the necessity of humans to be present in locations where the control action takes place (e.g., manufacturing lines in the space). Furthermore, more emphasis is currently placed on the control algorithm itself being located on the computing cloud (Control as a Service (CaaS)). Such solution, whilst providing undoubtful benefits in terms of cost, flexibility, ease of modifications, and maintenance, also poses certain problems which need to be addressed, for instance, resilience of control actions and security of information flow and information processing.

1.4. Cloud Computing. According to the NIST publication [12], cloud computing is defined as "a model that allows network access to computing resources (e.g., networks, servers, memory, applications, and services) that can be quickly delivered and based on the use of services provided by the service provider." The principle of operation is to transfer the burden of IT services from the local computer to the server with the possibility of permanent access through client computers [4]. This results in greater reliability, because regardless of what happens to the client computer, all services will continue to work.

Cloud-based control of systems and Control as a Service (CaaS) have become a focus of interest relatively recently. In [13], expectations and challenges related to such systems are outlined. In [14, 15], the term Control as a Service in relation to a programmable logic controller (PLC) used for industrial automation tasks (soft-PLC) is explored. The focus is on communication requirements and on the scalability of the controller.

1.5. Cyber-Security. In parallel, there is a growing interest in research on cyber-security of such configurations.

Standard, IT based methods of providing cyber-security are being proposed. Those include separation of business and production networks by strong passwords and firewalls.

In [16, 17], forms of encryption of information sent to and from the controller are proposed to eliminate possible cyber-attacks. Then, the design of the encryption mechanism in such a way that it would not be necessary to deencrypt the information in order to perform standard operations of the controller (addition and multiplication) is discussed.

In [18], a cyber-physical test-bed which can be used to test the Modbus TCP protocol in response to cyber-attacks is described. The test-bed uses a real-time power grid simulator. Some possible scenarios of cyber-attacks are discussed.

1.6. Contribution of This Paper. Our interest is in investigating whether the information about the process variables and the control actions could be additionally used to more precisely detect possible cyber-attacks. There exist well developed techniques of "fault detection and isolation," which are widely applied to continuous processes, especially in process industry (petrochemical, energy generation). Fault-related residuals are being detected on the basis of the analysis of process data. We propose to use similar approach to analyze robotic/manufacturing systems. Another group of methods which we intend to use is related to "control performance assessment and benchmarking" [19, 20]. In this approach, the changes in controller performance are detected as signalizing possible faults or cyber-attacks on the system. We propose to extend those techniques to cover mechatronic systems, characterized by a combination of discrete-event and continuous-event control.

Hence, the main focus and the main novelty proposed in this paper are in applying the above methods to robotic/mechatronic systems in order to test their ability to detect cyber-attacks in such systems.

For that purpose a laboratory stand is being prepared at Warsaw University of Technology which would emulate all major components of Industry 4.0 manufacturing mechatronic systems. In the subsequent sections, the concept and the build of this mechatronic stand are presented. Next, its cyber-physical functionalities are discussed, including communication links and discussion of nodes where cyber-attacks could penetrate. Further, possible configuration of the system to test effects of different cyber-attacks is discussed. Finally, some initial results are presented.

2. Description of the Experimental Set-Up

In the Institute of Automatic Control and Robotics of the Warsaw University of Technology, a laboratory stand was created, presenting issues related to Industry 4.0. The

conceptual diagram of the stand is presented in Figure 1, whereas Figure 2 shows a photo view of the actual equipment.

In general terms, the stand is provided with two operator panels used to define what is to be produced at the station and to monitor the work of the station. All control elements of the station are connected with each other and the whole system is connected to the Internet, which allows remote access to the station.

The stand explores the possibility of production-on-demand, personalized production.

With reference to Figure 3, product personalization means here the choice of which discs will be connected to each other; the lower disc with the RFID tag is deposited with the upper disc, whose upper surface is of a certain color. With the help of the operator panel, or remotely, the production scheme is defined, i.e., the operator selects which color of the upper disc is to be placed on the next product. The discs with RFID tags are issued by the station 1. The issued discs are placed in the indexing table seats. The vision system verifies the color on the disc in the inner socket (upper disc) and saves this information to the database. After the release of all the disks and their identification, the discs of the desired color are placed on the appropriate discs on the table. Next, the robot transfers the produced item (two discs) to the packing and further to the customer, according to the desired color specification. At the same time, the next pair of discs is assembled, possibly with different color.

The stand can be divided into two parts: part one is the FANUC cooperating robot and part two is the indexing table and manipulators for manipulating simple details (Figure 3). The PLCs responsible for controlling the laboratory and for the exchange of data between controllers, the robot, and the cloud-based platform are mounted on the industrial racks.

In addition to being a good representation of the production-on-demand, the stand also enables various tests and benchmarks related to the communication between the substations and cyber-security of the installation. This is facilitated by

(i) Using three different industrial controllers, which need to communicate among themselves as well as communicate with their assigned sensors and actuators

(ii) Using different ways of communication, which could be via direct digital link, or via Ethernet, or via wireless link through the cloud

(iii) Ability to remotely define the production plan and, moreover, to connect to the cloud for reprogramming of the controllers

There are three PLC controllers from different manufacturers: Allen-Bradley Compact GuardLogix 5380, Beckhoff CX5140, and Wago 750-880. The devices connected to them are a valve island from Festo, two control panels for Beckhoff and Allen-Bradley PLC controllers, and two servo drives (from Beckhoff and Allen-Bradley).

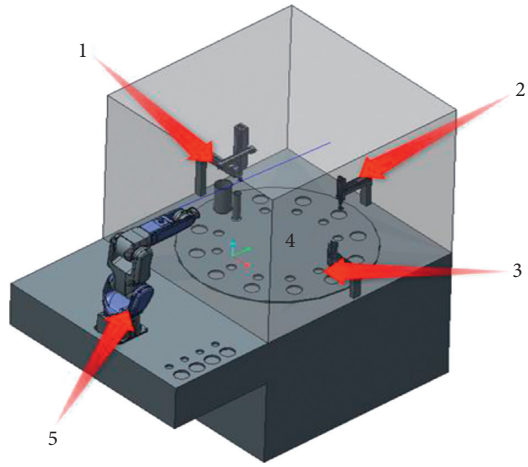


Figure 1: The concept of the set-up presenting the issues of Industry 4.0; pneumatic servo-manipulator, 1; pneumatic manipulators with vision system, 2; pneumatic manipulators, 3; the indexing table with slots for discs and RFID system, 4; the collaborating robot, 5.

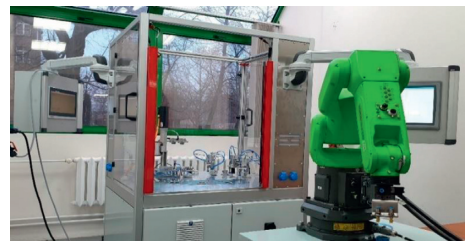


Figure 2: General view of the entire station.

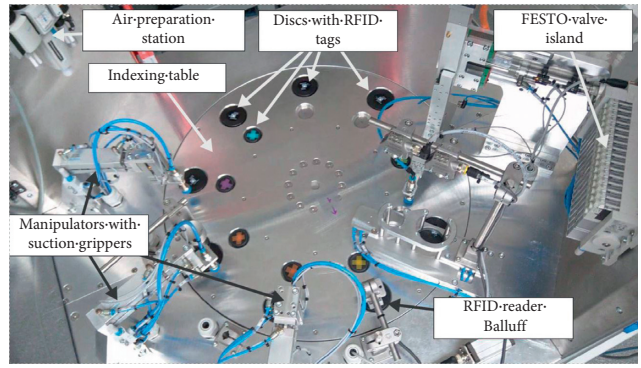


Figure 3: The indexing table with colored disks.

Controller Beckhoff is connected with both Allen-Bradley and Wago, whereas the latter two are not connected. In this way, Beckhoff becomes the main PLC, because it is able to synchronize all of them.

The above experimental stand enables tests of various algorithms related to the cyber-security. An example of a simple algorithm implemented on the stand will be presented below. The example will concern mainly one of the elements available at the stand, the pneumatic manipulator. The details of operation of such a manipulator are shown in Figure 4.

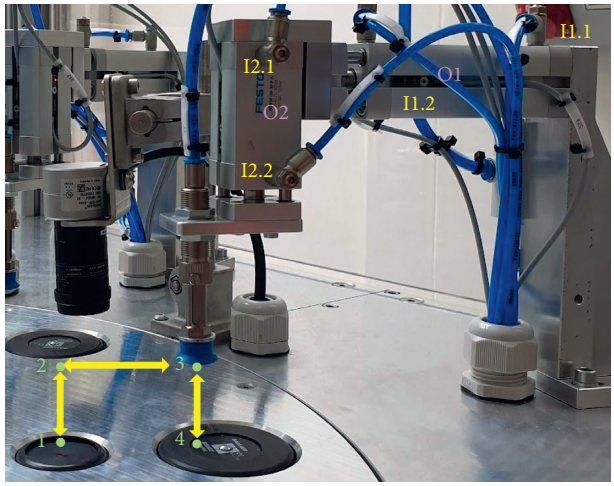


Figure 4: Pneumatic manipulator used on the stand.

The manipulator consists of a base, two pneumatic cylinders, and a pneumatic gripper mounted on the second cylinder. The actuators allow the gripper to move in two axes. In this situation, it is possible to operate the disks that are placed in the sockets of the indexing table. The position of the manipulator's actuators is controlled by pneumatic solenoid valves. Their state is changed via the digital output signals O1 and O2. Information about the current position of the manipulator is obtained from the limit sensors located in the end positions of each of the actuators (signals I1.1, I1.2, I2.1, and I2.2). The correct movement related to shifting the small colored disc on the large disc takes place between the marked points 1-2-3-4 and the return movement between points 4-3-2-1.

3. Description of Communication Links

3.1. Communication between the PLCs. Communication between PLCs should allow not only to easily exchange data (presumably also between networks) but also enable the synchronization of tasks and command drivers. For example, if one of the devices connected to the Wago driver successfully completes its task, it should be able to provide the Allen-Bradley driver with information about it, which at the same time will give the driver a signal that it should perform the next program point.

The Beckhoff TwinCAT 3 software is very comprehensive, which gives configuration options and many communication protocols and interfaces to choose from. In the version shown above, the Beckhoff CX5140 and Allen-Bradley Compact GuardLogix 5380 are connected via Ethernet/IP, while the Beckhoff CX5140 and Wago 750-880 are connected via Modbus/TCP.

Choosing two different network protocols will give the ability to test communication between protocols as well as the resistance of individual protocols to external interference and attacks.

Beckhoff controller is also connected to the cloud, which contains a database with detailed information about the operation of the system.

3.2. Application of Techniques of Industry 4.0. In developing this laboratory stand, it was important to use elements of Industry 4.0 such as communication via the Ethernet/IP standard, RFID tags, visualization using the HMI interface, remote, wireless stand management via VPN, and remote desktop. The devices have been connected and configured to exchange data with a local or a remote database.

(i) TCP/IP (Transmission Control Protocol/Internet Protocol) is a layered model of a set of protocols needed to send information. This model consists of four layers: application, transport, Internet, and network interface.

(ii) Ethernet/IP uses the Ethernet standard at the physical layer, encapsulating data according to UDP or TCP protocols at the transport layer and data link and implements Common Industrial Protocol (CIP) at the application layer in the network communication structure according to the simplified OSI model, TCP/IP [15].

(iii) Common Industrial Protocol (CIP) is a communication protocol designed for applications in industrial automation. It includes functions: control, security (CIP Safety), synchronization (CIP Sync), motion (CIP Motion), configuration, and information. CIP is independent of the method of data exchange, the network protocol, which allows easy data flow between different networks and thus also outside of Ethernet/IP.

(iv) Modbus/TCP: Modbus is used to transfer information between devices and allows remote programming of devices and supports distributed I/O. Modbus TCP is a Modbus standard adapted to the Ethernet standard [1]. Modbus TCP due to its openness and due to licensing is supported by many devices of different companies

(v) The RFID (Radio Frequency Identification) system consists of a read/write head and a data carrier (data carrier/RFID tag). The RFID system is used for remote recognition of objects, devices, and products. The RFID head consists of an antenna that sends and receives a signal. RFID tags are constructed from integrated circuit, antenna, and housing. Data carriers may be read-only (RO), write once read many (WORM), or rewrite (RW). Depending on the power supply methods, there are passive tags powered by the waves sent by the reader, active tags with battery power, and semi-passive tags with power only for the integrated circuit. In this laboratory stand, the tags are passive and have RW functions.

(vi) The IO-Link master (Master IO-Link-750-657 Wago) is a device used in IO-Link communication to exchange data between the controller and actuators or measuring devices called the IO-Link device. IO-Link is a common open communication standard (IEC 611131-9 standard).

Master IO-Link can connect to the controller using various protocols, e.g., Profibus, Profinet, Ethernet/IP, DeviceNet, EtherCAT, and CC-Link. The IO-Link masters from various companies can be used with all actuating/measuring devices that support this standard on the market. More specifically, IO-Link is described in [21].

3.3. Remote Management. Configuring the operation of the stand is partially performed remotely through the use of tunnel Virtual Private Network (VPN) L2TP/IPSEC. This enables remote connection and communication with those devices which are network connected. The controllers are programmed from the level of the desktop computer on which the PLC manufacturers' software has been installed. By using the Remote Desktop Protocol (RDP), it is possible to remotely operate that computer from the level of the graphical interface of Windows 10. With the help of a tunnel VPN, it is possible to connect to the computer and freely program the PLCs from any place where there is a fast enough access to the Internet and (thanks to the popularity of the protocol) from almost any device (laptops with Windows 10 and Linux and an Android smartphone were tested).

Regardless of the desktop, the Beckhoff CX5140 on which Windows 10 CE is installed also supports the RDP protocol and allows remote configuration.

Both the VPN and RDP connections are encrypted and password protected.

3.4. Data Flow in the System. Allen-Bradley and Wago controllers write data to the register of the Beckhoff controller. This register contains sensor states and program variable states, which are important for the functioning of the entire station. Data flow is illustrated in Figure 5.

In addition to the possibility of serving a larger number of workstations, the cloud also provides a certain degree of redundancy for stored data, providing security in their storage in case of failure.

The test stand has been connected to the Google Cloud database as part of a free limited test package. The MySQL instance has been launched; the other ready-made solutions also enable the launch of MS SQL and PostgreSQL instances. The configuration of the connection to the cloud database turned out to be the key element. When the system was built, it was decided to use a simple way to limit connections to a narrow pool of clients' IPs, thus creating basic authorization. Adding certificates (CA) to the configuration of encryption is not a major problem; however, it requires expansion of the system with additional Open Database Connectivity (ODBC) drivers locally and configuration of the encrypted connection. This configuration makes it significantly more difficult to start data acquisition using the TwinCAT system, because the system itself does not support Secure Sockets Layer (SSL).

Another potential solution that provides encryption and increased security is to use Secure Socket Shell (SSH) tunneling with machines in Google Cloud. That method uses private IPs, thereby increasing security compared to methods using public database instance IPs. The choice of a method and configuration of the connection is critical for securing data flow. The latter described method shows the highest security potential by implementing the following elements:

(i) Allowing disabling access to the database via public IP

(ii) Allowing running an encrypted SSL connection using certificates of authenticity (CA)

(iii) Virtual Private Cloud (VPC) is used for connecting local parts to the cloud system

(iv) Cloud database IP is only available from the private network

Adding a database in the cloud complicates the system and requires an open connection to the external network. Its implementation requires a thorough design of the network architecture as well as the implementation of all possible cyber-security measures to protect against unauthorized interference from outside.

4. Possible Configurations of the Tests of Cyber-Security

The configuration of the laboratory, the components used, and the wide spectrum of communication channels available enable various tests related to resilience of control algorithms. The following could be possible reasons for this manufacturing line not to work properly:

(i) Malfunction of one of the controllers, for instance, the RFID tagged discs are not placed properly on the table, or the robot is not collecting the produced items on time

(ii) Falsification of readings of sensors, for instance, the color determined by the camera

(iii) Errors in communication between the PLCs, hence the information about the combination of the RFID code and the color of the second disc may not be properly transmitted; also, information about completing stage 1 may not reach the controller responsible for stage 2

(iv) Errors and falsification of information transmitted remotely (via remote terminal and/or from the cloud, for instance, the production demand for the next element, or the colors of the discs stored in the database

For each of the described anomalies, the reasons could be in the faults developed within the system itself or, indeed, could result from cyber-attacks.

The set-up enables testing different configurations, e.g.,

(i) Some parts are connected via Ethernet, some by direct cable digital link

(ii) Remote access form the Internet to certain parts of the system can be set up

(iii) Some of the communication channels are duplicated by other means of communication, or opposite

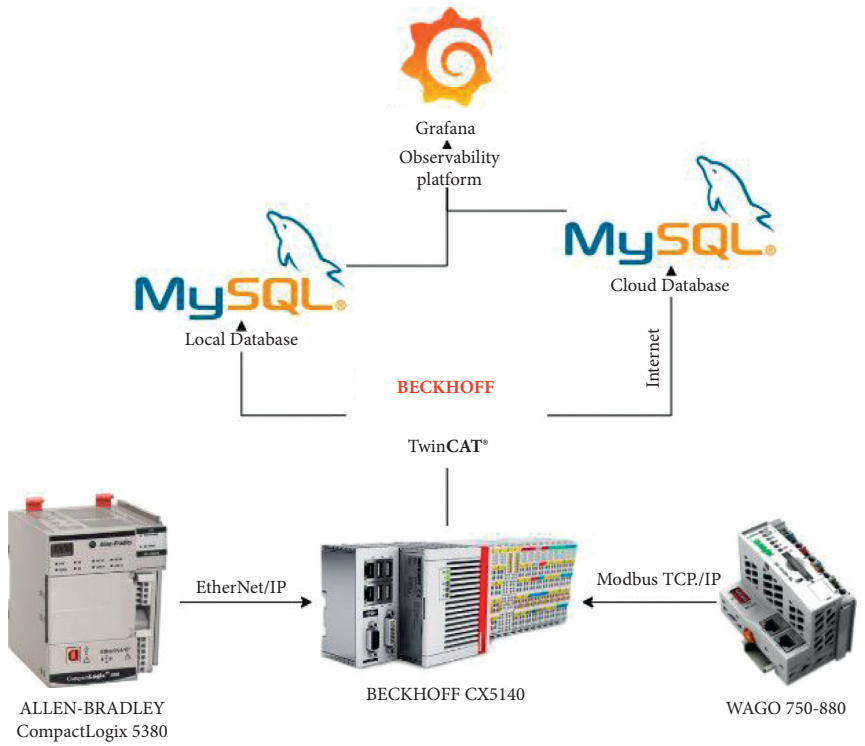


Figure 5: Data flow in the system.

(iv) Some of the devices are designed not to communicate with others

(v) The PLC controller(s), one or more, can be replaced by a soft-PLC located on the cloud

Resilience of the control structure, in different configurations, could be assessed.

5. Initial Results

The controller service program has been developed, which enables simulation of a flexible production process. With the help of switches, it is possible to simulate failure and/or execution of one product unit. Depending on the color of the ordered discs, other relevant signals are issued, which in turn simulate the launch of different machines in certain modes of operation.

Figure 6 presents the algorithm of operation for the station when production-on-demand is carried out. Firstly, for each RFID tag of the lower (bigger) discs, a corresponding desired color of the upper disc is determined from the operator panel or from the remote control panel and saved (locally or in the cloud) in the production schedule. The manipulators should properly place the colored discs so that, at the end, each of the RFID (lower) discs has on top of it a smaller (colored) disc with a color as selected by the user.

According to the program, if the controller is not working, it goes into standby mode. There should be no production downtime in the actual production process, so it can additionally be assumed that after all personalized orders are completed, the process goes to mass production of some standard catalog units.

The cloud communication support program can be divided into two parts. The first part is responsible for collecting data from the controller and sending them to the cloud. The second part is responsible for receiving data from the application in the cloud. Visualization of the entire data flow is shown in Figure 7 (prepared in Node-RED application).

In order to send data to the cloud, two blocks should be used: an input block to read data from the controller's memory and an output block to transfer this data to the cloud.

Initial tests/experiments have been performed on the stand. Due to the high complexity of the whole system, only a simple case of possible cyber-attack on the pneumatic actuator is discussed in this paper. The work of the actuator has been described earlier, in Section 2 of the paper.

The algorithm proposed below for the detection of potential attacks is based on the analysis of the current state of digital inputs (information from the limit switches). Table 1 shows the possible digital input states for the manipulator positions for the correct disk ratio. This set of acceptable signal values in each step will be used in the algorithm as a set of reference data.

The developed algorithm, based on residual's generation, consists in comparing the current state of digital inputs (the current position of the actuators) with the reference state, which is presented in Table 1. Additionally, the current state is used in the next step as the previous state and these two states are used for comparison. This allows not only to determine whether a given state is correct but also whether the sequence of steps is correct and whether the manipulator correctly shifts the disc. When a given

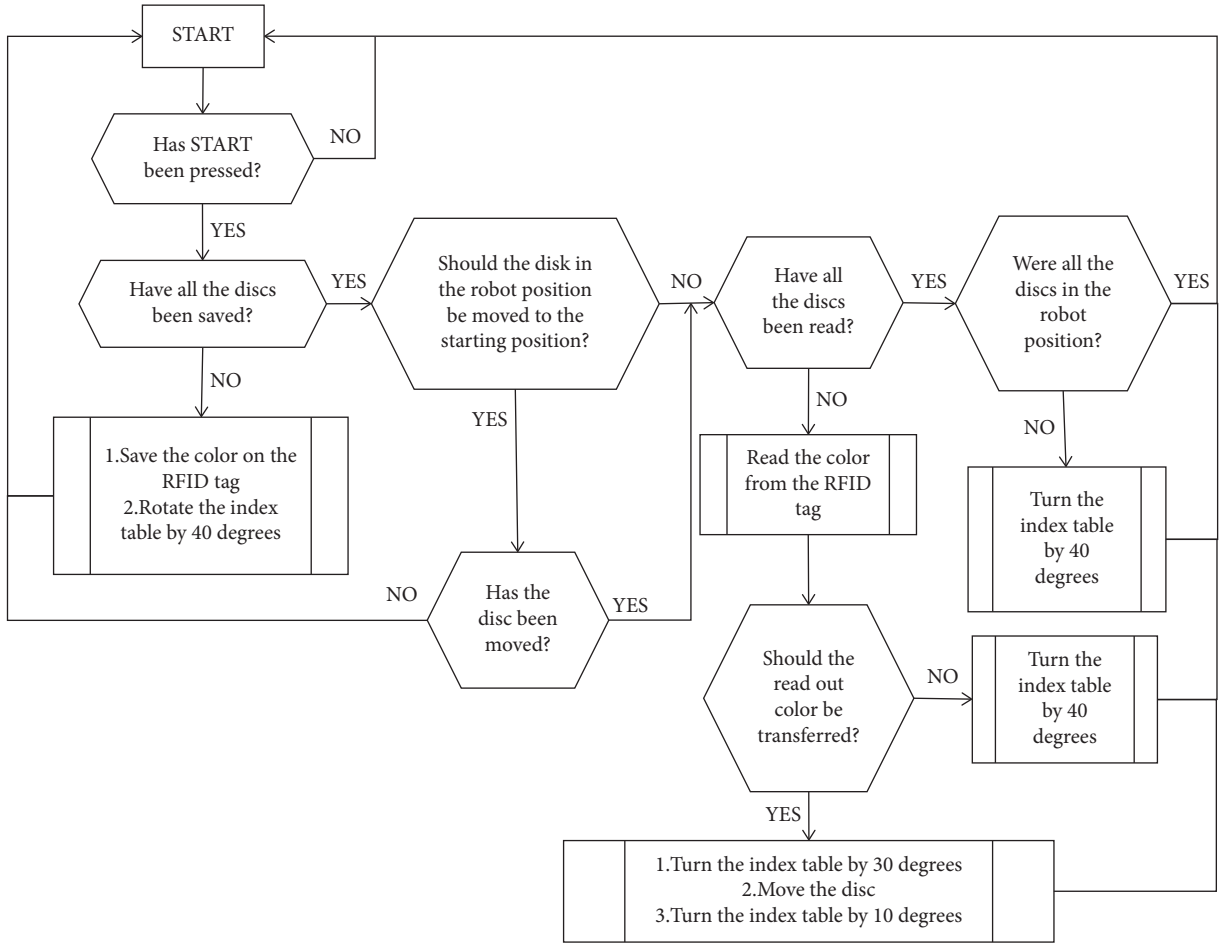


Figure 6: Algorithm of operation of the station.

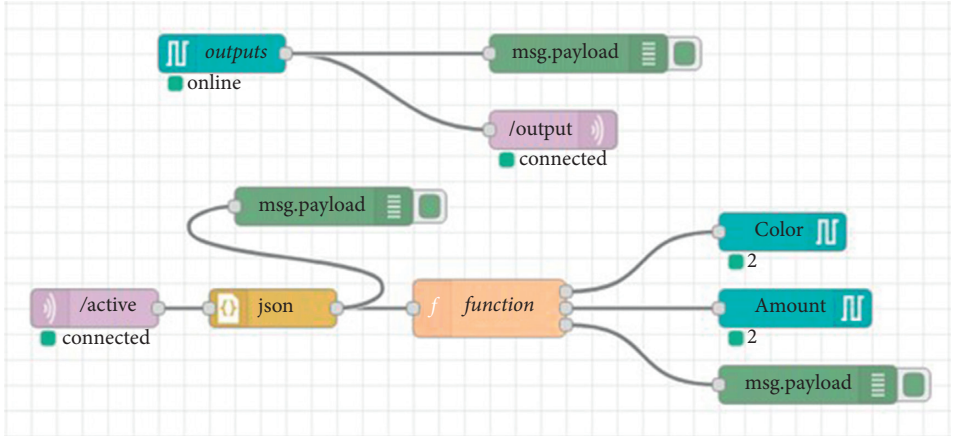


Figure 7: Block diagram of the cloud communication support program.

state is not identified among the correct states, or if the state is correct but is not executed in the correct order, then it may be suspected that there has been an external interference in the operation of the manipulator. Such interference can be, for example, a cyber-attack.

Figure 8 shows a block diagram of the algorithm implemented to detect potential cyber-attacks.

As a result of the algorithm's operation, four possible conditions of the manipulator's operation are identified:

(1) The state (as indicated by received signals) is correct, no undesirable change of signals occurred

(2) The state is correct and has not changed compared to the state from the previous step (such information

Table 1: Correct state of digital signals for particular positions of the manipulator.

Nr.	I1.1	I1.2	I2.1	I2.2
1	0	1	0	1
2	0	1	1	0
3	1	0	1	0
4	1	0	0	1
3	1	0	1	0
2	0	1	1	0
1	0	1	0	1

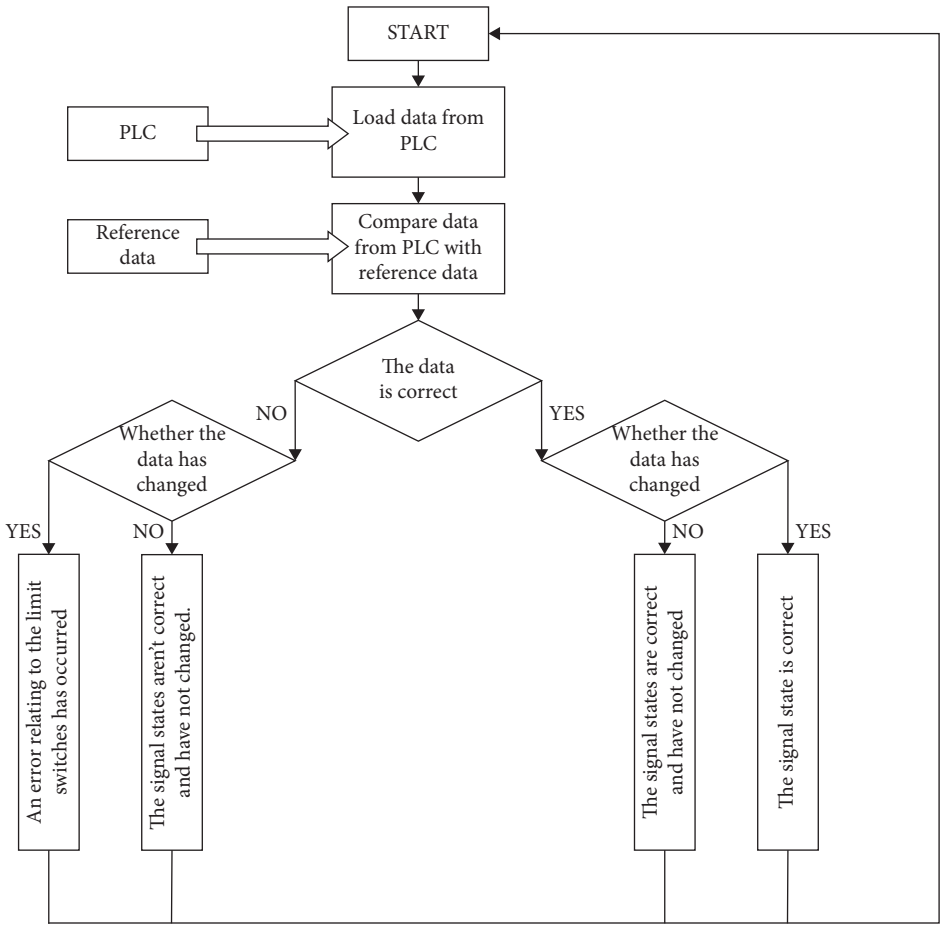


Figure 8: Scheme of the algorithm for detecting changes in the operation of the manipulator.

was introduced due to possible slower changes of signals than the operation of the algorithm itself)

(3) The state is incorrect, an undesirable change in the manipulator's operation

(4) The signal condition is not correct and has not changed from the previous step

The algorithm developed in this way was implemented and tested on the stand described above. An example of the algorithm's performance is shown in Figure 9.

The above example demonstrates that the algorithm based on calculation of residuals can be used for detection of possible malfunctions of a manufacturing system, caused for instance by cyber-attacks.

While testing the operation of the system, several problems have been identified and analyzed.

The first problem is that incorrect operation of one controller or incorrect setting of elements for which this controller is responsible may cause an error on another controller or prevent the stand from continuing to work at all. An example of such a situation is the absence of the RFID tag in the appropriate position, which would prevent further operation of the program because it would wait indefinitely for reading information from the tag. Pneumatic cylinders can be another example of this when they are not in the home position. Moving the actuators from their position prevents correct operation of the controller's logic, because signals which are sent from the limit switches attached to the

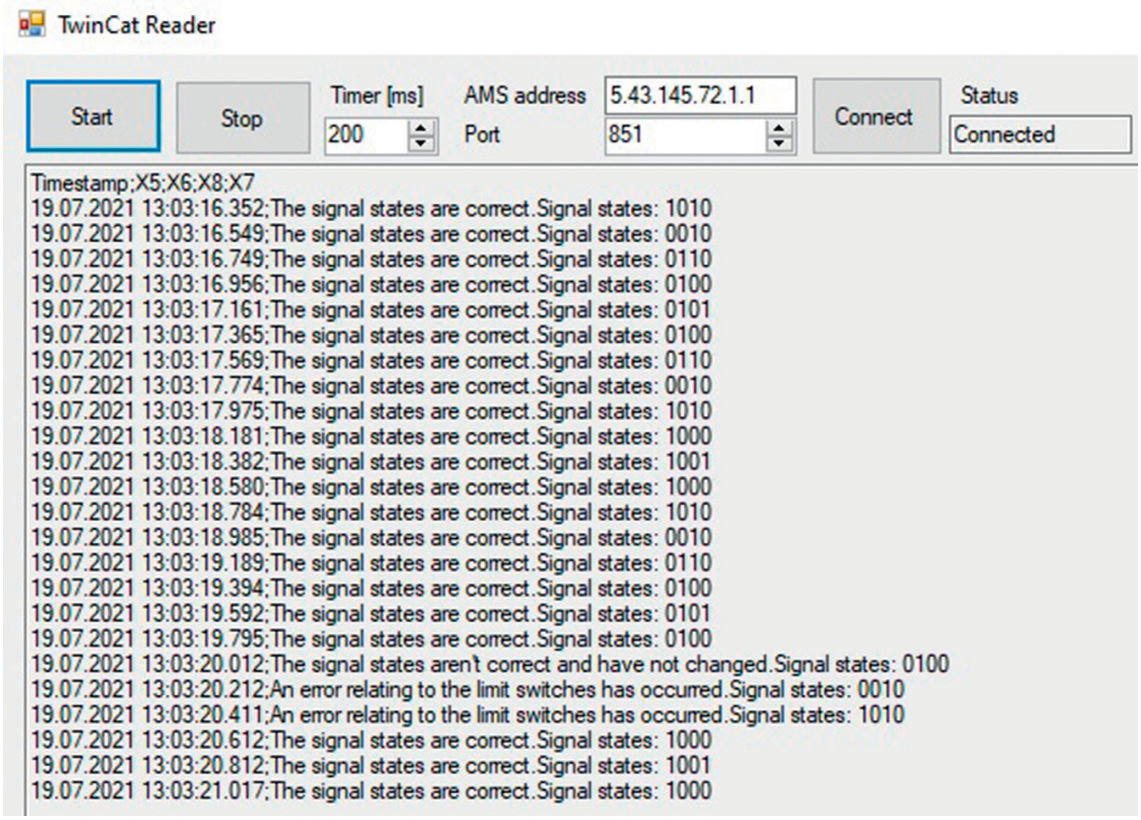


Figure 9: An example window showing the results of the algorithm.

actuators in this case do not allow proceeding to the next stage of the program.

Another problem that has been noticed when programming the drivers is that of not sending the values of variables that contain information about completing the task properly. This problem can occur if communication between the controllers is not working properly. In this situation, the movements of actuators can be made a number of times that is different than required by the program's logic. This problem is minimized due to the use of industrial communication protocols which use a three-way handshake to establish communication and use checksums guarantee packets which are sent in full.

The stability of the data acquisition system has been tested in the event of the database connection being interrupted. A problem has been encountered in the form of difficulties in resuming records after regaining connection stability.

There have also been tests of the speed of downloading data from various instances of the database (cloud and local) by running a refresh of information in the Grafana environment every one second. Simple queries gave satisfactory results, while more complex queries could result in queues of many queries.

The stand can be set-up and controlled remotely. Hence, we invite colleagues to collaborate on testing of cyber-attacks, algorithms for detection and elimination of such attacks, and/or algorithms for resilient control of the system under attacks.

6. Conclusions

This paper presents a new laboratory stand, devoted to design of Industry 4.0 enabled production systems and to testing their resilience to cyber-attacks. The main components of the laboratory have been presented with emphasis on possible connections and ways in which subsystems communicate. Configurations of the tests for cyber-security have been discussed.

Furthermore, some results of tests performed on the stand, utilizing the methodology of assessing the values of residua, have been presented. Based on the above, factors influencing the detection of possible cyber-attacks have been discussed.

As possible directions of future work, we would like to consider the following:

(i) Distinguishing between the cyber-attacks and the faults of equipment. Redundancy in the communication channels and a possibility to duplicate the control actions; by running them locally on the controllers as well as on the cloud would facilitate such a task. In the current configuration, the controller code is hosted directly on the controller. However, it could be duplicated on the cloud and the results of the two can be compared.

(ii) Design the control/sequencing algorithms in such a way that even in cases of some malfunctions/attacks the system can still operate or, at least, it can stop

operating smoothly and safely, without producing wrong products and causing any damage. Many methods and tools have been discussed in the literature in the context of fault tolerant control. They could be a useful starting point for such investigations.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

Andrzej Ordys acknowledges support from the National Agency of Academic Exchange (NAWA), "Polish Returns," Grant No. PPN/PPO/2018/1/00063/U/00001. Krzysztof Kukielka, Andrzej Ordys, Jan Maciej Kościelny, Michał Syfert, Paweł Wnuk, Jakub Możaryn, and Bartłomiej Fajdek acknowledge support from the POB Research Centre Cybersecurity and Data Science of Warsaw University of Technology within the Excellence Initiative Program-Research University (ID-UB).

References

[1] P. Wittbrodt and I. Łapuńka, *Przemysł 4.0-Wyzwanie Dla Współczesnych Przesiębiorstw Produkcyjnych, Innowacje W Zarządzaniu I Inżynierii Produkcji, Tom II*, Oficyna Wydawnicza PTZP, Opole, Poland, 2017.

[2] D. Gerwin, "Manufacturing flexibility: a strategic perspective," *Management Science*, vol. 39, no. 4, 1993.

[3] S. Wang, J. Wan, D. Li, and C. Zhang, "Implementing smart factory of industrie 4.0: an outlook," *International Journal of Distributed Sensor Networks*, vol. 2016, Article ID 3159805, 2016.

[4] J. Schlechtendahl, F. Kretschmer, A. Lechler, and A. Verl, "Communication mechanisms for cloud based machine controls," in *Proceedings of the 47th CIRP Conference on Manufacturing Systems*, Elsevier, Windsor, Canada, April 2014.

[5] M. Olszewski, "Mechatronizacja produktu i produkcji–przemysł 4.0," *Pomiary Automatyka Robotyka*, vol. 20, no. 3, pp. 13–28, 2016.

[6] X. Krasniqi and E. Hajrizi, "Use of IoT technology to drive the automotive industry from connected to full autonomous vehicles," *IFAC-PapersOnLine*, vol. 49, no. 29, pp. 269–274, 2016.

[7] M. A. R. Garcia, R. Rojas, L. Gualtieri, E. Rauch, and D. Matt, "A human-in-the-loop cyber-physical system for collaborative assembly in smart manufacturing," *Procedia CIRP*, vol. 81, pp. 600–605, 2019.

[8] A. R. Patel, R. S. Patel, N. M. Singh, and F. S. Kazi, "Vitality of robotics in healthcare industry: an internet of things (IoT) perspective," in *Internet of Things and Big Data Technologies for Next Generation Healthcare*, C. Bhatt, N. Dey, and A. Ashour, Eds., vol. 23, pp. 91–109, Studies in Big Data, Springer, Berlin, Germany, 2019.

[9] B. Satuyeva, C. Sauranbayev, I. A. Ukaegbu, and H. S. V. S. K. Nunna, "Energy 4.0: towards IoT applications in Kazakhstan," *Procedia Computer Science*, vol. 151, pp. 909–915, 2019.

[10] D. Mocrii, Y. Chen, and P. Musilek, "IoT-based smart homes: a review of SystemArchitecture, software, communications, privacy and security," *Internet of Things*, vol. 1-2, pp. 81–98, 2018.

[11] A. Verl, A. Lechler, S. Wesner et al., "Armin lechler, stefan wesner, andreas kirstädter, jan schlechtendahl, lutz schubert, sebastian meier, an approach for a cloud-based machine tool control," in *Proceedings of the Forty Sixth CIRP Conference on Manufacturing Systems 2013*, Elsevier, Setubal, Portugal, May 2013.

[12] P. Mell and T. Grance, *The NIST Definition of Cloud Computing*, Recommendations of the National Institute of Standards and Technology, Gaithersburg, Maryland, 2011.

[13] L. Monostori, "Cyber-physical production systems: roots, expectations and R&Dchallenges," in *Proceedings of the 47th CIRP Conference on Manufacturing Systems*, Windsor, Canada, April 2014.

[14] T. Goldschmidt, M. K. Murugaiah, C. Sonntag, B. Schlich, S. Biallas, and P. Weber, "Cloud-based control: a multi-tenant, horizontally scalable soft-PLC," in *Proceedings of the 2015 IEEE 8th International Conference on Cloud Computing*, pp. 909–916, New York, NY, USA, July 2015.

[15] J. Schlechtendahl, F. Kretschmer, Z. Sang, A. Lechler, and X. Xu, "Extended study of network capability for cloud based control systems," *Robotics and Computer-Integrated Manufacturing*, vol. 43, pp. 89–95, 2017.

[16] F. Farokhi, I. Shames, and N. Batterham, "Secure and private cloud-based control using semi- homomorphic encryption," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 163–168, 2016.

[17] K. Junsoo, C. Lee, H. Shim et al., "Encrypting controller using fully homomorphic encryption for security of cyber-physical systems," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 175–180, 2016.

[18] B. Chen, N. Pattanaik, A. Goulart, K. L. Butler-purry, and D. Kundur, "Implementing attacks for modbus/TCP protocol in a real-time cyber physical system test bed," in *Proceedings of the 2015 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR)*, pp. 1–6, Charleston, SC, USA, May 2015.

[19] A. W. Ordys, D. Uduehi, and M. Johnson, "Process control performance assessment, from theory to implementation," *Monograph Series: Advances in Industrial Control*, Springer Verlag London, Berlin, Germany, 2007.

[20] A. Ordys and M. J. Grimble, "Benchmarking and tuning PID controllers," in *PID Control in the New Millennium: Lessons Learned and New Approaches*, V. Visioli, Ed., Springer-Verlag, Berlin, Germany, 2012.

[21] F. Almada-Lobo, "The industry 4.0 revolution and the future of manufacturing execution systems (MES)," *Journal of Innovation Management*, vol. 3, no. 4, pp. 16–21, 2015.

Hindawi
Complexity
Volume 2021, Article ID 6626842, 23 pages
https://doi.org/10.1155/2021/6626842

Research Article

Applied Mechatronics: Designing a Sliding Mode Controller for Active Suspension System

Aydin Azizi[1] **and Hamed Mobki**[2]

[1]*School of Engineering, Computing and Mathematics, Oxford Brookes University, Wheatley Campus, Oxford OX33 1HX, UK*
[2]*Department of Engineering, German University of Technology in Oman, Muscat, Oman*

Correspondence should be addressed to Aydin Azizi; aydin.azizi@brookes.ac.uk

Received 19 November 2020; Revised 24 March 2021; Accepted 24 April 2021; Published 7 May 2021

Academic Editor: Matilde Santos

The suspension system is referred to as the set of springs, shock absorbers, and linkages that connect the car to the wheel system. The main purpose of the suspension system is to provide comfort for the passengers, which is created by reducing the effects of road bumpiness. It is worth noting that reducing the effects of such vibrations also diminishes the noise and undesirable sound as well as the effects of fatigue on mechanical parts of the vehicle. Due to the importance of the abovementioned issues, the objective of this article is to reduce such vibrations on the car by implementing an active control method on the suspension system. For this purpose, a conventional first-order sliding mode controller has been designed for stochastic control of the quarter-car model. It is noteworthy that this controller has a significant ability to overcome the stochastic effects, uncertainty, and deal with nonlinear factors. To design a controller, the governing dynamical equation of the quarter-car system has been presented by considering the nonlinear terms in the springs and shock absorber, as well as taking into account the uncertainty factors in the system and the actuator. The design process of the sliding mode controller has been presented and its stability has been investigated in terms of the Lyapunov stability. In the current research, road surface variations are considered as Gaussian white noise. The dynamical system behavior for controlled and uncontrolled situations has been simulated and the extracted results have been presented. Besides, the effects of existing uncertainty in the suspension system and actuator have been evaluated and controller robustness has been checked. Also, the obtained quantitative and qualitative compressions have been presented. Moreover, the effect of controller parameters on the basin of attraction set and its extensiveness has been assessed. The achieved results have indicated the good performance and significant robustness of the designed controller to stabilize the suspension system and mitigate the effects of road bumpiness in the presence of uncertainty and noise factors.

1. Introduction

In recent decades, the automotive industry has witnessed rapid progress, and several studies in the cases of design of car shape based on aerodynamic optimization [1], [2], optimization of the air intake system of the engine [3], investigation of sound quality for passenger car [4], stress analysis and design improvement of door hinge for compact cars [5], study of fuel consumption for various driving styles in conventional and hybrid electric vehicles [6, 7], design optimization of the cowl cross bar-light [8], investigation of a rear independent suspension for light vehicle [9], and study of nonlinear control of suspension system [10–12] have been conducted to develop and optimize different aspects of vehicle performance. The suspension system is one of the main components of a car which plays an important role in providing passenger comfort. The suspension system is generally divided into three classes of passive, semiactive, and active. The main task of this system is to reduce the amplitude and unwanted effects of vibrations imposed by road surface roughness on the car's chassis. In addition to providing comfort, reducing such impacts plays an important role in the transportation of goods and military industries. For example, when transporting fruits, animals, or delicate materials, vibrations with large amplitudes are a matter of concern. Reducing vibrations is also essential for

accurate aiming of military vehicles. Furthermore, damped vibrations improve vehicle steering [13], braking performance, and energy consumption. Additionally, fatigue can be prevented in many parts of the vehicle such as the body and engine.

As the most effective method for reducing the impact of chassis vibrations, active suspension has been extensively studied in the literature. Rizvi et al. [14] have utilized robust control techniques for active control of a car suspension system. Wang et al. [15] presented a new model-free fractional-order sliding mode control based on an Extended State Observer (ESO) for quarter-car active suspension systems.

Mardani [16] has investigated energy harvesting, handling, and ride comfort tradeoff between passive and active suspension systems, using a PID controller. Khodadadi and Ghadiri [17] have utilized PID, fuzzy logic, and Hinf controllers to analyze the suspension system. Kumar et al. [18] applied a fractional-order fuzzy PD controller to analyze the car suspension system. In [19], the adaptive neurofuzzy inference system control has been utilized to analyze the active suspension system. Other control strategies related to the control and optimization of industrial systems that can be also applied to the control of the suspension system have been presented in [20], [21].

Nonlinear factors and parameters of spring and the shock absorber are the main points to be taken into consideration. Furthermore, unmodeled uncertainties in the suspension and other mechanical systems can make it difficult to control the system. In addition, the actuator can also include uncertainty or defects, where defects are inherently considered a type of uncertainty. The abovementioned issues, that is, nonlinearity, uncertainty, and disturbances, complicate the design procedure of the active suspension or controller.

To reduce the effects of uncertainty and disturbances, robust control methods are needed since not all control methods are capable of providing stabilization [22–24]. Sliding mode controller and Hinf are two of the most important robust control methods offering a favorable performance in reducing the effects of noise and uncertainty [25–27]. It is necessary to determine the norms related to uncertainties and disturbances, in the Hinf method. But for the SMC method, it is only necessary to have the upper and lower bands of these terms. Also, in the Hinf method, there is no problem called chatting phenomenon but in the SMC method, there is such a phenomenon [28, 29]. However, with procedures, the effects of this phenomenon can be reduced. In general, considering that, in this study, only the upper and lower bands of uncertainties and disturbances are available, the sliding mode method has been utilized to control. It is noteworthy that, in the recent study [30], the SMC method was used to design the controller and the Hinf method was used to determine the optimal feedback gain. This study shows that the SMC and Hinf methods are combined and used simultaneously to control the dynamical systems [31].

Given the importance of controlling and modeling the car suspension system, this study discusses the control of this system and the evaluation of its stability by means of a basin attraction set. As mentioned before, in addition to the linear term, springs and shock absorbers include nonlinear terms. Furthermore, these parts as well as actuators can also include uncertainties. Therefore, for the precise simulation of the system, all of the abovementioned parameters are considered in this work. Due to the high capability of the sliding mode method in controlling the nonlinear systems and overcoming the effects of uncertainty and disturbance, this method is considered to control it. In addition to the discussion of structure control, the discussion of the basin of attraction set is also examined at the end of the results section. Moreover, the effects of uncertainty in the system and actuator together with the effects of controller parameters on the dynamical behavior of the system and the region of attraction set are investigated. In this paper, based on the authors' previous related research [31–34], road surface variations are considered as Gaussian white noise. Obviously, the use of fuzzy-sliding mode [35], neural-sliding mode, or second-order sliding mode controller along with more complexity and calculations can increase the efficiency of the control system or reduce the effects of chattering. In addition to these cases, it is even possible to increase the scope of research by considering the existence of faults in the structure or investigating the fault-tolerant problem in the presence of time-delayed Markov jump [36] and examine the control of the faulty system. However, considering that the main purpose of this paper is reducing uncertainty effects and considering nonlinear factors in dynamic modeling and evaluating stability, so only the common first-order sliding mode scheme is recruited to analyze of results.

2. Model Description

Before presenting the controller design and mathematical molding of the quarter-car suspension system, the geometric and physical model of this system is presented here.

Considerable research has been conducted by assuming the suspension system in the linear range [37]. However, in a more precise point of view, considering the actual situation of spring and damper between car and the tire [38], as well as considering that springs and dampers include nonlinear terms, it is more accurate to consider these nonlinear terms in simulation procedure. Furthermore, tire stiffness consists of a set of linear and nonlinear terms. Therefore, in this paper, to increase the accuracy of the physical model, nonlinear terms of springs and damper also participate in physical modeling.

Figure 1 shows the nonlinear quarter-car model used in this study. As shown, the system includes two masses of M_1 and M_2. Sprung mass M_1 represents a quarter of the equivalent mass of the vehicle and passengers, while the unsprung mass M_2 represents the mass of the wheel. As shown in the figure, a spring, a shock absorber, and an actuator are present between masses M_1 and M_2.

Depending on how the spring and shock absorber are linked between the tire and car, these parts may be considered with nonlinear terms. One such model is the Machperson model [38]. In addition, as mentioned earlier,

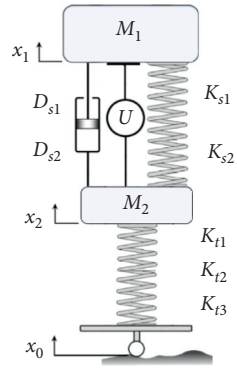


Figure 1: Schematic view of active quarter-car suspension model.

springs and shock absorbers naturally include nonlinear terms; although these nonlinear terms may have a diminutive effect, considering them increases the simulation accuracy.

The spring and shock absorbers consist of linear and nonlinear where K_{s1} and D_{s1} represent the constants for the linear terms of the spring and shock absorber, respectively. Moreover, $K_{s2}$and D_{s2} express the nonlinear spring and shock absorber coefficients. The actuator provides the force needed for controlling the system. The tire exhibits a stiffness and, therefore, is modeled as a spring connected to mass M_2 from the top, while its other end is in contact with the road surface from the bottom. The coefficient K_{t1} represents the constant for the linear term of the tire stiffness, while the coefficients K_{t2} and K_{t3} denote the constants for its nonlinear terms. The vertical displacement of masses M_1 and M_2 is also shown and indicated by x_1 and x_2, while x_0 represents the vertical variations in the pavement.

3. Mathematical Modeling

3.1. Governing Dynamic Equation of Quarter-Car System. The governing dynamic equation of the system shown in Figure 1 can be represented as follows:

$$\begin{aligned} &M_1\ddot{x}_1 + K_{s1}(x_1 - x_2) + K_{s2}(x_1 - x_2)^3 + D_{s1}(\dot{x}_1 - \dot{x}_2) + D_{s2}(\dot{x}_1 - \dot{x}_2)^2 + f_1(x,t) = h(x,t)u \\ &M_2\ddot{x}_2 - K_{s1}(x_1 - x_2) - K_{s2}(x_1 - x_2)^3 - D_{s1}(\dot{x}_1 - \dot{x}_2) - D_{s2}(\dot{x}_1 - \dot{x}_2)^2 \\ &+K_{t1}(x_2 - x_0) + K_{t2}(x_2 - x_0)^2 - K_{t3}(x_2 - x_0)^3 + f_2(x,t) = -h(x,t)u, \end{aligned} \tag{1}$$

where $\dot{x}_1$ and $\dot{x}_2$ represent the time derivatives of x_1 and x_2 and u is the actuator force, in which its magnitude is specified by the controller. Given that actuators are not usually considered to be ideal and include uncertainty, the bounded function $h(x,t)$ was considered to model the actuator uncertainty. Note that the minimum value of $h(x,t)$ is positive and denoted by h_0. Functions $f_1(x,t)$ and $f_2(x,t)$ also signify the unmodeled uncertainty of the system that has a known upper limited value.

3.2. Sliding Mode Controller Design. In this section, the sliding mode controller has been designed to stabilize the vertical movement of mass M_1. The goal is to stabilize the vertical movement of M_1, that is, x_1, and to track x_d. To this end, the tracking error is considered as $e = x_1 - x_d$ [39]. The tracking error is also expressed by $\dot{e} = \dot{x}_1 - \dot{x}_d$. Assuming $e = e_1$, the dynamic of tracking error can be expressed as

$$\begin{cases} e = x_1 - x_d, \\ \dot{e} = \dot{x}_1 - \dot{x}_d, \end{cases} \quad \text{Or} \quad \begin{cases} \dot{e}_1 = e_2, \\ \dot{e}_2 = \ddot{x}_1 - \ddot{x}_d. \end{cases} \tag{2}$$

By determining $\ddot{x}_1$ from equation (1) and substituting it in equation (2), the error dynamics can be written as

$$\begin{cases} \dot{e}_1 = e_2, \\ \dot{e}_2 = \dfrac{1}{M_1}\left[-K_{s1}(x_1 - x_2) - K_{s2}(x_1 - x_2)^3 - D_{s1}(\dot{x}_1 - \dot{x}_2) - D_{s2}(\dot{x}_1 - \dot{x}_2)^2 - f_1(x,t) + h(x,t)u\right] - \ddot{x}_d \end{cases}. \tag{3}$$

Note that the second relation of equation (1) is used to determine the value of x_2 in each time step. In an actual system, there is no need to solve such relations for determining x_2, because its values are directly obtained from the installed sensors. However, since this study is a simulation, the second relation in equation (1) must be solved. Moreover, the vertical variations of the road are considered as white Gaussian noise; this section briefly discusses the stochastic Itô and Stratonovich integrals.

The Itô and Stratonovich integrals are explained in the following given their application in solving stochastic differential equations. These two integrals are widely used to solve such equations in cases where the studied system is subjected to the Wiener process. Consider $\int_0^T h(t)\mathrm{d}w$, where

w: $[0,T]\times\Omega \longrightarrow R$ represents a Wiener process and X: $[0,T]\times\Omega \longrightarrow R$ is a semimartingale compatible with filtration. Under these conditions, and relying on the Riemann Sum, the above integral value can be expressed in two ways:

$$\sum_{j=0}^{N-1} h\left(t_j\right)\left[w\left(t_{j+1}\right)-w\left(t_j\right)\right], \tag{4}$$

$$\sum_{j=0}^{N-1} h\left(\frac{t_j+t_{j+1}}{2}\right)\left[w\left(t_{j+1}\right)-w\left(t_j\right)\right]. \tag{5}$$

Equations (4) and (5) represent the integral forms of Itô and Stratonovich relations, respectively, where N is the number of divisions in the time interval $[0,T]$ and j is the counter. $\mathrm{d}t = T/N$ represents the time interval between two samples. It must be noted that the Wiener process is the integral of white noise; in other words, $x_0\mathrm{d}t = \mathrm{d}w$.Since the term $K_{t1}x_0\mathrm{d}t$ represents the term $h(t)\mathrm{d}w$ and the value of K_{t1} is constant with respect to time, Itô and Stratonovich assume identical forms according to equations (4) and (5) [40].

Having discussed the Itô and Stratonovich integrals, the design of the sliding mode controller is explained in the following. Considering $e_1 = x_1 - x_d$ and $e_2 = \dot{x}_1 - \dot{x}_d$ and by placing $x_1 = e_1 + x_d$ and $\dot{x}_1 = e_2 + \dot{x}_d$ in equation (3), the dynamic error equation can be rewritten as follows:

$$\begin{cases} \dot{e}_1 = e_2, \\ \dot{e}_2 = \dfrac{1}{M_1}\left[-K_{s1}\left(e_1+x_n\right) - K_{s2}\left(e_1+x_n\right)^3 - D_{s1}\left(\dot{e}_1+\dot{x}_n\right) - D_{s2}\left(\dot{e}_1+\dot{x}_n\right)^2 - f_1(e,x,t) + h(e,x,t)u\right] - \ddot{x}_d \end{cases}, \tag{6}$$

where $x_n = x_d - x_2$ and $\dot{x}_n = \dot{x}_d - \dot{x}_2$.

To design the sliding mode controller, we consider the sliding surface s and its time derivative as follows:

$$\begin{aligned} s &= e_2 + \lambda e_1, \\ \dot{s} &= \dot{e}_2 + \lambda\dot{e}_1 = \dot{e}_2 + \lambda e_2. \end{aligned} \tag{7}$$

By inserting e_2 and $\dot{e}_2$ from equation (6) into (7), we have

$$\dot{s} = \frac{1}{M_1}\left[-K_{s1}\left(e_1+x_n\right) - K_{s2}\left(e_1+x_n\right)^3 - D_{s1}\left(\dot{e}_1+\dot{x}_n\right) - D_{s2}\left(\dot{e}_1+\dot{x}_n\right)^2 - f_1(e,x,t) + h(e,x,t)u\right] - \ddot{x}_d + \lambda e_2. \tag{8}$$

3.2.1. Lyapunov Function and Controller Stabilization:. In this section, we shall discuss the Lyapunov function and how to stabilize the controller using the time derivative of the Lyapunov function [41–43].

Definition 1. The Lyapunov point $\overline{x}$ is considered stable if there exists $\varepsilon > 0$ such that $\delta(\varepsilon) > 0$ and $\|x(0) - \overline{x}\| < \delta$; then, $\|x(t) - \overline{x}\| < \varepsilon$ for all $t > 0$. Note that the initial condition $\delta(\varepsilon)$must exist for $\varepsilon > 0$.

Definition 2. The fixed point $\overline{x}$ is asymptotically stable if it is stable and there exists $\eta > 0$ where $\lim_{t\longrightarrow\infty}\|x(t) - \overline{x}\| = 0$ whenever$\|x(0) - \overline{x}\| < \eta$.

The Lyapunov stability and asymptotic stability are shown in Figures 2(a) and 2(b), respectively. As shown in Figure 2(a)), in Lyapunov's stability, $\|x(t) - \overline{x}\|$ lies in the surrounded space and remains in this region for different times. Note that, in this case, the expression $\|x(t) - \overline{x}\|$ is not necessarily required to reach zero as time approaches ∞. However, as shown in Figure 2b), $x_1(t)$ moves toward $\overline{x}$ over time, and as $t \longrightarrow \infty$, the value of $\|x(t) - \overline{x}\|$ approaches zero.

For Lyapunov stability, $\delta(\varepsilon)$ actually represents the radius of basin attraction set $\overline{x}$. The aforementioned concepts help better understand the design and function of the sliding mode controller. In the absence of uncertainty, noise, and disturbance, asymptomatic stability is achievable for deterministic systems using sliding mode controllers. However, these factors are inherently present in dynamic systems, making it impossible to have asymptotic stability, and merely Lyapunov stability would be achievable. In deterministic systems, we can implement all the specifications accurately to ensure asymptotic stability, and even the convergence rate of the dynamic error can be specified. However, in presence of uncertainty or noise in the system, asymptomatic stability cannot be guaranteed. As shown in equations (1) and (8), the values for parameters of $h(x,t)$, $f_1(x,t)$, $f_2(x,t)$, and x_0 are unpredictable and thus cannot be accurately modeled in the controller. Therefore, asymptomatic stability for this system may not be guaranteed, and only the Lyapunov stability can be achieved.

In the following, a definition is given for Lyapunov's function that its existence implies the existence of system stability. Naturally, it should be noted that the lack of such function is not a reason for instability. Actually, the existence of such a term is a sufficient condition for stability. Although this definition may not provide a straightforward method for determining the control law, but based on this definition, the control law will be presented to stabilize the feedback control system.

Definition 3. Consider the energy function or Lyapunov function V such that $V > 0$. If $\dot{V} < 0$, then the stability of this system is guaranteed.

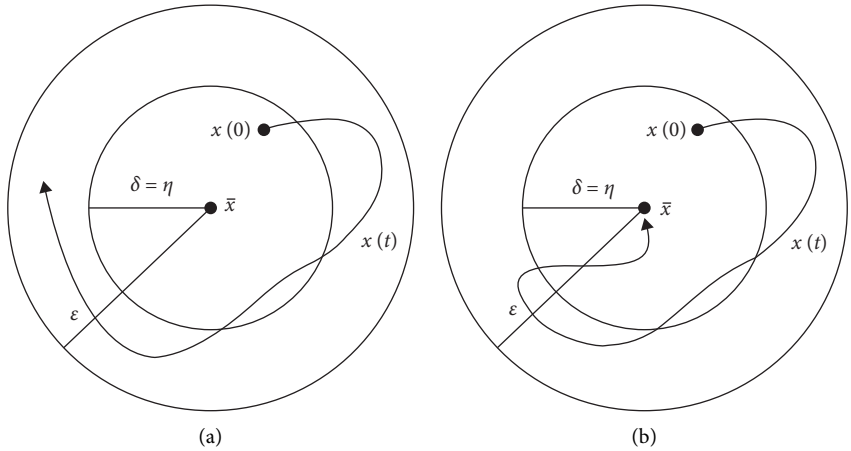


Figure 2: (a) Lyapunov stability and (b) asymptotic stability [44].

In fact, $\dot{V} < 0$ indicates the decrescent rate of the energy function, which indicates a decrease in energy and tendency of V toward zero (for $\dot{V} > 0$, the energy of the system gradually increases and the system goes toward instability). In the continuation of this section, according to the above definition, we will determine the control law for the feedback control system with the Lyapunov stability approach.

The considered energy function is $V = (1/2)s^2$, which represents a positive function. The time derivative of this function is equal to $\dot{V} = s\dot{s}$. Now, if we can satisfy the condition $\dot{V} = s\dot{s} < 0$, the control feedback rule may be presented according to Definition 3.

By inserting $\dot{s}$ from equation (8) in $\dot{V} = s\dot{s}$, we have

$$\dot{V} = s\left\{\frac{1}{M_1}\left[-K_{s1}\left(e_1 + x_n\right) - K_{s2}\left(e_1 + x_n\right)^3 - D_{s1}\left(\dot{e}_1 + \dot{x}_n\right) - D_{s2}\left(\dot{e}_1 + \dot{x}_n\right)^2 - f_1(x,t) + h(x,t)u\right] - \ddot{x}_d + \lambda e_2\right\}. \tag{9}$$

Now for stability, we have to determine u in such a way that the above equation remains negative.

Determining u through this relationship is complex. Therefore, in the following, we define an upper limit for $\dot{V}$. If the seminegative condition for this upper limit is met, then the condition $\dot{V} < 0$ is also met. To determine u, we assume that the whole expression to the right of this function is equal to $-\eta h(x,t)|s|$ where η is a positive number. According to $\eta > 0$, $h(x,t) > 0$, and $|s| > 0$, $-\eta h(x,t)|s|$ and $\dot{V}$ become negative and the condition of stable feedback control is guaranteed. In fact, this equation is the basis for determining control law and all calculations made in equations (12) and (13) are to determine this law:

$$\begin{aligned}\dot{V} &\le \frac{|s|h(x,t)}{M_1}\left\{\frac{1}{h_0}\left[K_{s1}\left|e_1 + x_n\right| + K_{s2}\left|e_1 + x_n\right|^3 + D_{s1}\left|\dot{e}_1 + \dot{x}_n\right| + D_{s2}\left(\dot{e}_1 + \dot{x}_n\right)^2 + F_1 + \frac{\left|\ddot{x}_d\right|}{M_1} + \frac{\lambda}{M_1}\left|e_2\right|\right]\right\} \\ &+ \frac{sh(x,t)u}{M_1} = -\eta h(x,t)|s| \\ \dot{V} &\le -\eta h(x,t)|s|,\end{aligned} \tag{10}$$

where $F_1 = \sup\{|f_1(x,t)|\}$ and $h(x,t) \ge h_0 > 0$ and η represents a positive number. The positive function F can then be defined as follows:

$$F = \frac{1}{M_1 h_0}\left[K_{s1}\left|e_1 + x_n\right| + K_{s2}\left|e_1 + x_n\right|^3 + D_{s1}\left|\dot{e}_1 + \dot{x}_n\right| + D_{s2}\left(\dot{e}_1 + \dot{x}_n\right)^2 + F_1 + \frac{\left|\ddot{x}_d\right|}{M_1} + \frac{\lambda}{M_1}\left|e_2\right|\right]. \tag{11}$$

Inequality (10) can be rewritten as follows:

$$\dot{V} \le |s|Fh(x,t) + s\frac{h(x,t)u}{M_1} = -|s|\eta h(x,t). \tag{12}$$

Considering the right side of (12) and dividing it by $h(x,t)$, the following relation is obtained:

$$\begin{aligned} s\frac{u}{M_1} &= -|s|\eta - |s|F, \\ u &= -M_1(\eta + F)\frac{|s|}{s} = -M_1(\eta + F)\text{sign}(s). \end{aligned} \tag{13}$$

The above equation indicates the control rule obtained based on the sliding mode method, where the function $\text{sign}(s)$ is defined as follows:

$$\text{sign}(s) = \begin{cases} 1, & s > 0, \\ 0, & s = 0, \\ -1 & s < 0. \end{cases} \tag{14}$$

Equations (11) and (13) together with the relation of $s = \dot{e}_1 + \lambda e_1$ are the main blocks to design the sliding mode controller. To simplify the feedback process and the basis for determining the control law; the method of determining control law and the control process is presented in Figure 3. As it turns out, the most complex part of control design is determining the function of F. To determine this term in each simulation and in each time step, the values of x_1, x_2, $\dot{x}_1$, and $\dot{x}_2$ are taken from the car model (using sensors in the real system). Also, $\dot{x}_d$ and $\ddot{x}_d$ are obtained based on reference input or x_d. Also, the sliding surface of s can be obtained using blocks of λe_1 and $\mathrm{d}e_1/\mathrm{d}t$. Finally, according to the sign of sliding surface and using $-M_1 F\text{sign}(s)$ and $-M_1\eta\text{sign}(s)$, the control force is determined, and it is imposed to the suspension system. As it is known, the design terms consist of λ and η where they should be determined by the designer. Another factor that is very important in the dynamic behavior of the system is uncertainty. There is also another parameter of σ that will be discussed later, which will be used in the form of $s/(|s| + \sigma)$ and instead of $\text{sign}(s)$ to reduce the chatting phenomenon. Therefore, in the results section, the effects of these parameters on the dynamic behavior of the suspension system will be fully investigated.

In the continuation of this part, the effects of controller parameters of convergence speed of controller will be discussed. If the error is considered as $\tilde{x} = x - x_d$, the sliding surface equation can be reflected in the form of $s(x,t) = ((\mathrm{d}/\mathrm{d}t) + \lambda)^{n-1}\tilde{x}$. In this article $n = 2$. Also for $n = 3$, the sliding surface is presented in the form of $s = \tilde{x} + 2\lambda\dot{\tilde{x}} + \lambda^2\tilde{x}$. Given the initial conditions $\tilde{x}(0) = e(0)$, the equation for the sliding surface represents a linear differential equation with constant coefficients. Based on [45], the tracking error tends to zero exponentially and with a time constant of $(n-1)/\lambda$, where for $n = 2$, it is equal to $1/\lambda$. This time constant is for the case where the initial error is placed on the sliding surface. If the initial position of the error is not on the sliding surface, it reaches the surface in less than $|s(t=0)|/\eta$ and then starts sliding on this surface. Therefore, the parameters of λ and η are the most important parameters in the study of convergence speed and their effect on the dynamic behavior of the system will be addressed in the continuation of this work.

Of course, the important point is that, due to the existence of white Gaussian noise, the tracking error or the position of the system will never be settled at zero. This state is like an example of a forced vibration with stochastic nature.

In the above section, the design method of a sliding mode controller was explained based on the Lyapunov function. In the following section, some notes are discussed on designing a sliding mode controller for a dynamic system that is only affected by white noise. Consider the sliding mode surface as follows [46]:

$$\dot{s} = M(x,t) + B(x,t)u + H(x,t)\dot{v}, \tag{15}$$

where $\dot{v}$ expresses the Wiener process. Under these conditions, instead of the above equations, the definition of the Lyapunov function can be alternatively used for a stochastic system as the range of the basin attraction set can be more conveniently determined. Note that in this case, we could consider the Lyapunov function as $V = (1/2)E[s^2]$ instead of $V = (1/2)s^2$, where $E[\bullet]$ indicates the expected value. In this case, instead of $\dot{V}$, we must use $\mathrm{d}V = (1/2)E[\mathrm{d}(s^2)]$, where $\mathrm{d}V$ represents infinitesimal variations in V. Moreover, contrary to the case considered in this study assuming $(\mathrm{d}/\mathrm{d}t)[(1/2)(s^2)] = 2s\dot{s}$, in the new situation, the following equation based on Ito's definition of derivation must be used for determining the infinitesimal changes in s^2:

$$\mathrm{d}\left(s^2\right) = 2s\mathrm{d}s + \mathrm{d}s\mathrm{d}s. \tag{16}$$

Further information can be found in [47, 48]. Considering that no information is available on the expected values $E[x_2]$ and $E[\dot{x}_2]$, this method may no longer be used to design a sliding mode controller, since x_2 and $\dot{x}_2$ are not necessarily a random Wiener process or white noise.

At the end of this section and before presenting the results, we are interested in presenting the importance of conservatism in controller design. One of the important topics in designing a controller based on the definition of Lyapunov function is conservatism in the presence of uncertainties. Reducing such a factor can increase the quality of system control or decrease the controller force. Considerable studies have been done in this field, which can be stated as the Linear Matrix Inequality (LMI) and parameter-dependent Lyapunov function methods [49]. The use of the LMI method to reduce the conservatism of linear systems is presented in [50]. One of the key features of the LMI method in reducing conservatism is its ability to be generalized to nonlinear systems and its easy implementation in MATLAB software [51]. Another study that has been done in this field is based on the quadratic Lyapunov function [49]. Other investigations in this case and based on the definition of Lyapunov function are presented in [49]. In this study, due to the presence of uncertainty, reducing the conservative can improve the controller design or decrease the control force. However, due to the fact that the study of this topic in this paper

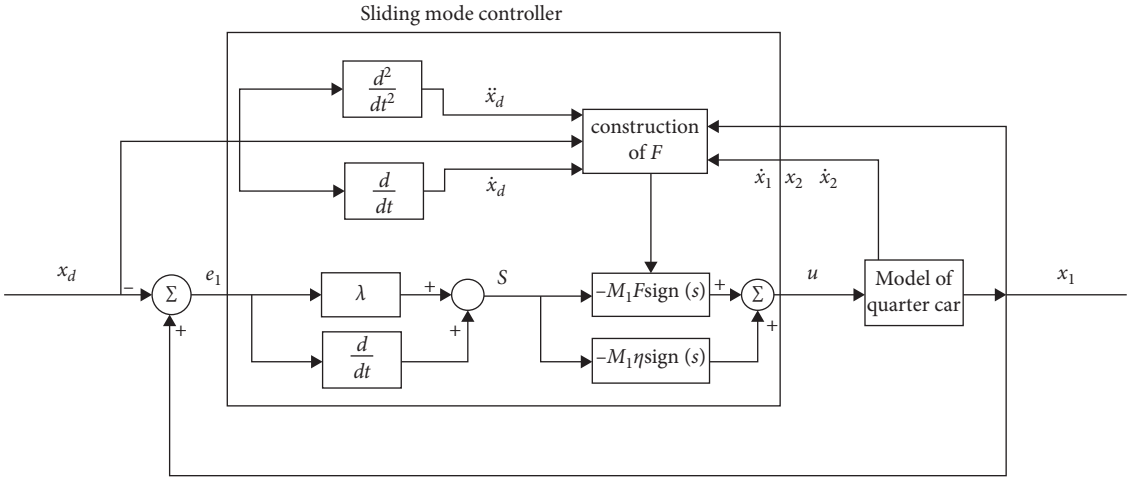


Figure 3: Block diagram of determining feedback control.

Table 1: Suspension parameters [51].

Suspension parameters	Value
Sprung mass M_1	295 kg
Unsprung mass M_2	39 kg
Linear damping coefficient D_{s1}	3482 Ns/m
Nonlinear square damping coefficient D_{s2}	580 Ns/m^2
Linear spring stiffness coefficient K_{s1}	15302 N/m
Nonlinear square spring stiffness coefficient K_{s2}	2728 N/m
Linear tire stiffness coefficient K_{t1}	60063 N/m
Nonlinear square tire stiffness coefficient K_{t2}	42509 N/m
Nonlinear cube tire stiffness coefficient K_{t3}	22875 N/m

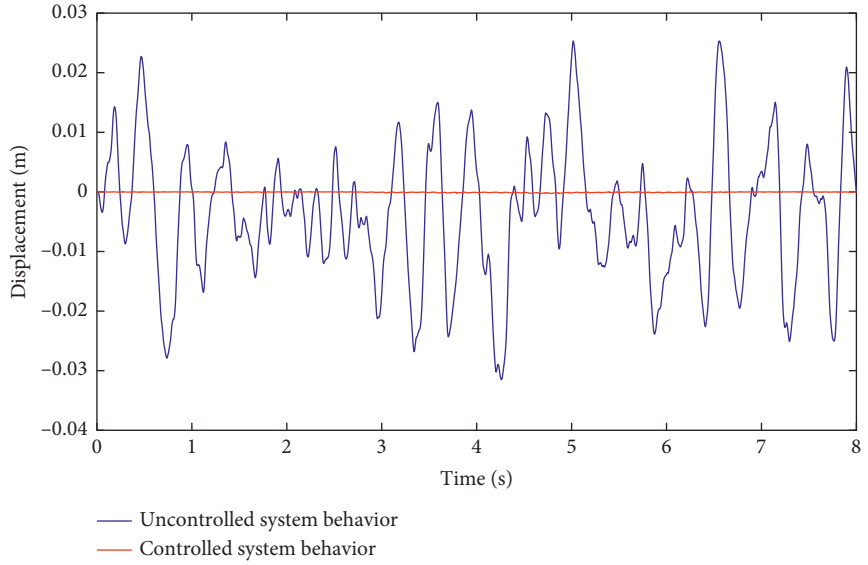


Figure 4: Uncontrolled and controlled system behavior using designed sliding mode controller for $\lambda = 1$ and $\eta = 10$.

causes much complexity and increases the unconventional scope of the paper, it is generally referred to here. Due to the significant importance of reducing conservatism, this issue should be studied in future studies for the design of controllers in nonlinear dynamic systems in the presence of white Gaussian noise.

4. Results and Discussions

4.1. Control of Quarter-Car System. In this section, we intend to present and discuss the extracted results. The specifications of the studied system are presented in Table 1 according to the ISO 2631–1:1997 standard [51]. To

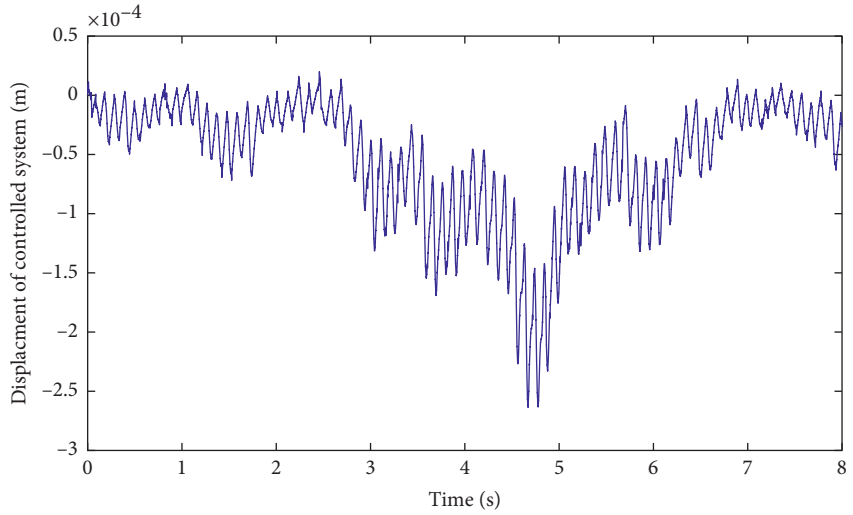
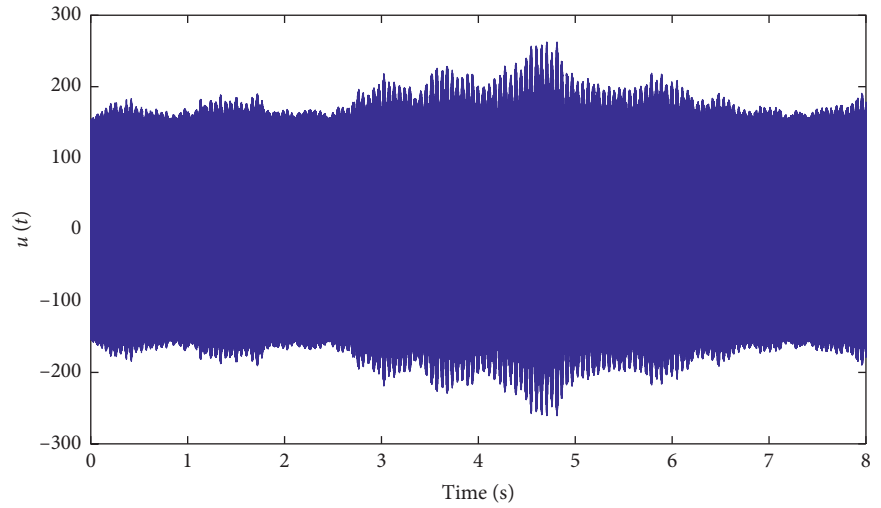
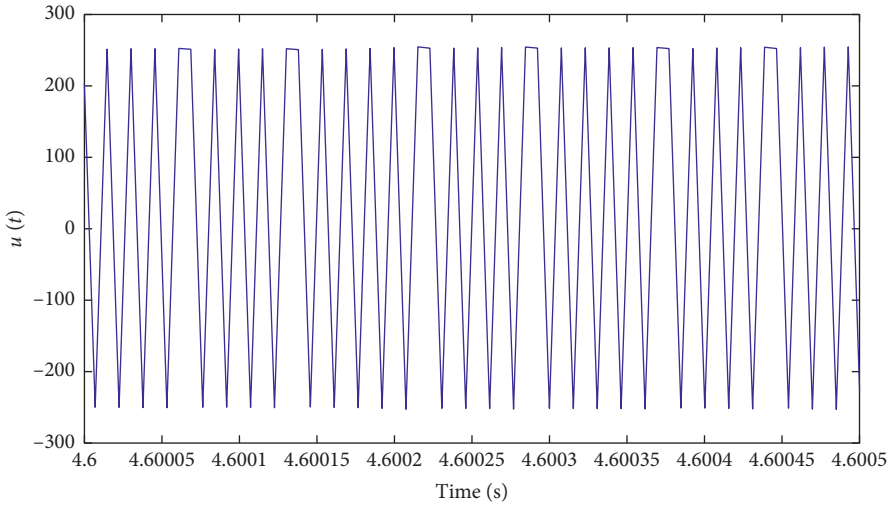


Figure 5: Controlled system behavior using designed sliding mode controller for $\lambda = 1$ and $\eta = 10$.

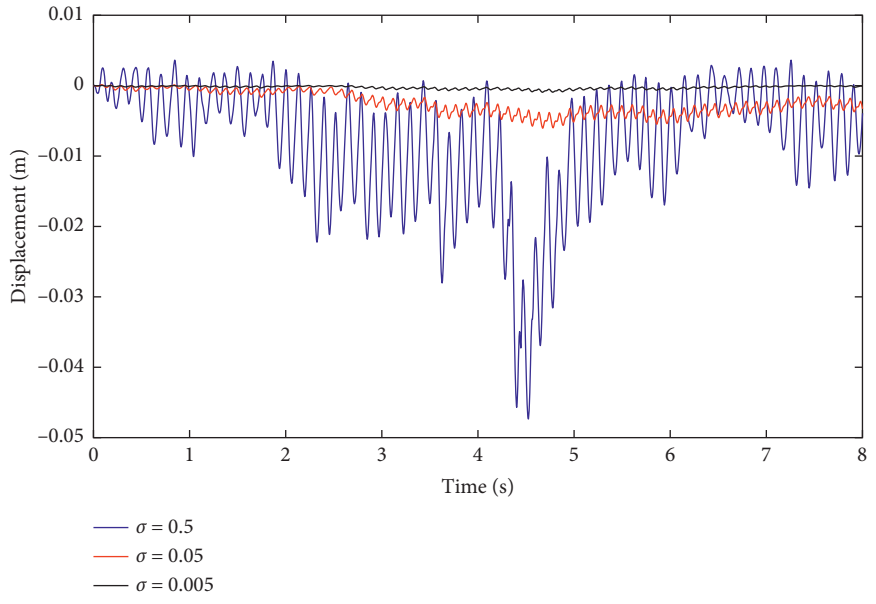


(a)

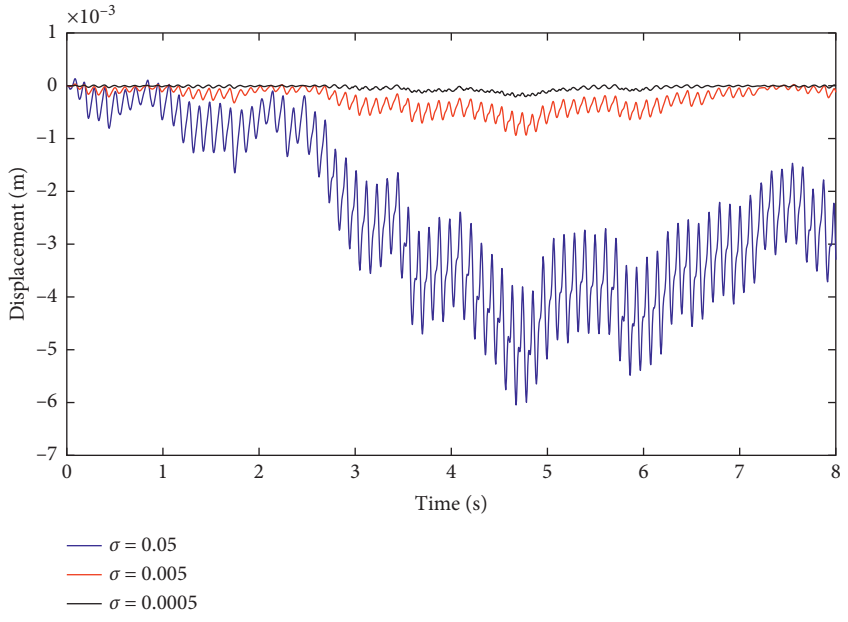


(b)

Figure 6: Variation of a control parameter of $u(t)$ for Figure 5. (a). $u(t)$. (b) Detailed view of Figure 6(a).



(a)



(b)

Figure 7: Controlled system behavior for different magnitudes of σ and $\lambda = 1$, $\eta = 10$. (a) (b)

Table 2: Maximum displacement versus parameter of σ.

σ	0.5	0.05	0.005	0.0005
Maximum displacement	0.047	0.006	0.00094	0.00021

obtain the results, x_d is considered as a constant with zero value.

Figure 4 shows the behavior of vertical movement of mass M_1 in the controlled and uncontrolled cases for $\lambda = 1$ and $\eta = 10$. As shown in the figure, the use of a sliding mode controller significantly reduces the amplitude of the vibration. Furthermore, the behavior of the active system is negligible compared to the passive system. Figure 5 shows only the behavior of the active system.

As mentioned, the chattering phenomenon is a major problem of the sliding mode controller. Given the function $\text{sign}(s)$ in the control force equation, the amount of this

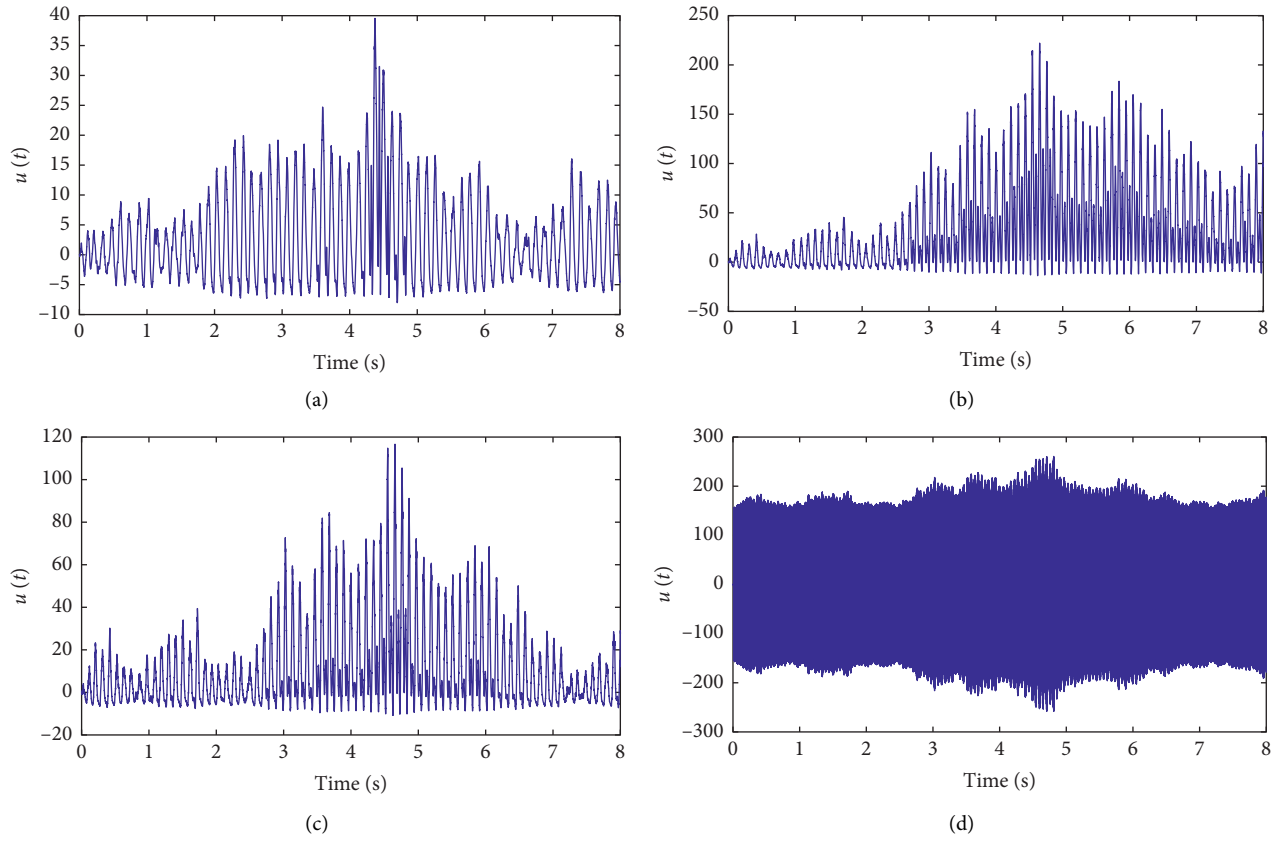


Figure 8: Variation of the control parameter of $u(t)$ for Figure 7. (a). $\sigma = 0.5$, (b). $\sigma = 0.05$, (c). $\sigma = 0.005$, and (d). $\sigma = 0.0005$.

Table 3: Maximum amount of control force $u(t)$for various amount of σ.

σ	0.5	0.05	0.005	0.0005
Maximum amount of $u(t)$	39.55	222.10	116.61	260.62

force changes its sign at a high frequency, which is due to the fact that sign $(s)$is discontinuous at $s = 0$. The changes in the control force presented in Figure 5 are depicted in Figure 6. As shown, the control force changes sign at a high frequency.

Note that the chattering phenomenon damages the actuators and reduces its lifecycle. Therefore, most designs of the sliding mode controller try to somehow eliminate the chattering phenomenon. As mentioned before, the function sign (s) is responsible for the chattering phenomenon because of discontinuity at point zero. To eliminate chattering, this article uses $s/(|s| + \sigma)$instead of sign (s), where σ is a positive and small number mathematically defined as follows:

$$\varphi_\sigma(s) = \begin{cases} 1, & |s| > \sigma, \\ \sigma^{-1}s, & |s| \le \sigma. \end{cases} \tag{17}$$

The function $\varphi_\sigma(s)$ is an approximation of sign (s) and is not its exact equivalent. The accuracy of this approximation is increased for smaller σ values.

Figure 7 shows the results of chattering reduction for different values of σ.

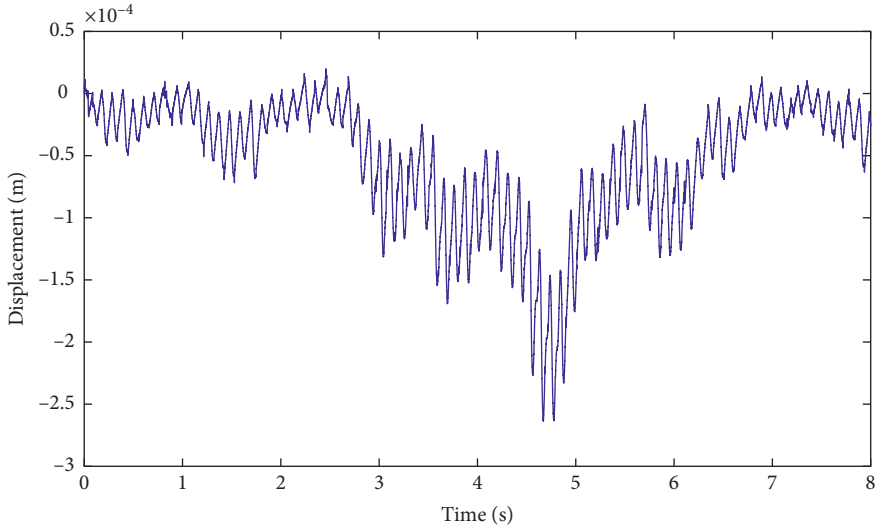
The maximum displacements shown in Figure 7 are presented in Table 2. As can be seen, with a decrease of σ, the level of maximum displacement decreases.

Moreover, variations of control forces for different values of $\sigma$are presented in Figure 8. As shown in Figure 7, by decreasing σ, the vibration amplitude is decreased. As shown in Figures 8(a)–8(c), the use of the function $\varphi_\sigma(s)$ eliminates the chattering phenomenon. However, as shown in Figure 8(d), the chattering phenomenon is still present for $\sigma = 0.0005$, suggesting that chattering cannot be eliminated for very small σ values. Therefore, for proper operation of the controller and reduction of the vibrations, the values $\sigma = 0.05$ and $\sigma = 0.005$ are suitable, as they reduce the vibration amplitude and eliminate the chattering phenomenon. Note that, however, for $\sigma = 0.5$, chattering is not detected, and according to Figure 7(a), the amplitude of vibrations is significant in this situation.

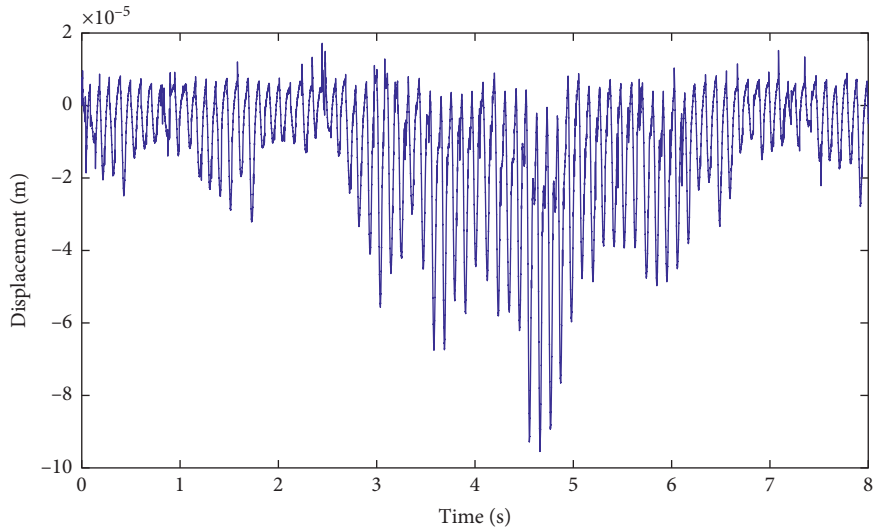
Also, the maximum amount of control force $u(t)$ for various amounts of σ is presented in Table 3. As it can be seen from this table, the maximum amount of $u(t)$ increases with the decrease of σ.

In this section, we shall discuss the impact of changes in λ and η on the behavior of the system. Figure 9 shows the behavior of the system for various λ values when $\eta = 10$. As it can be seen, the maximum amplitude of the system behavior decreases as λ increases.

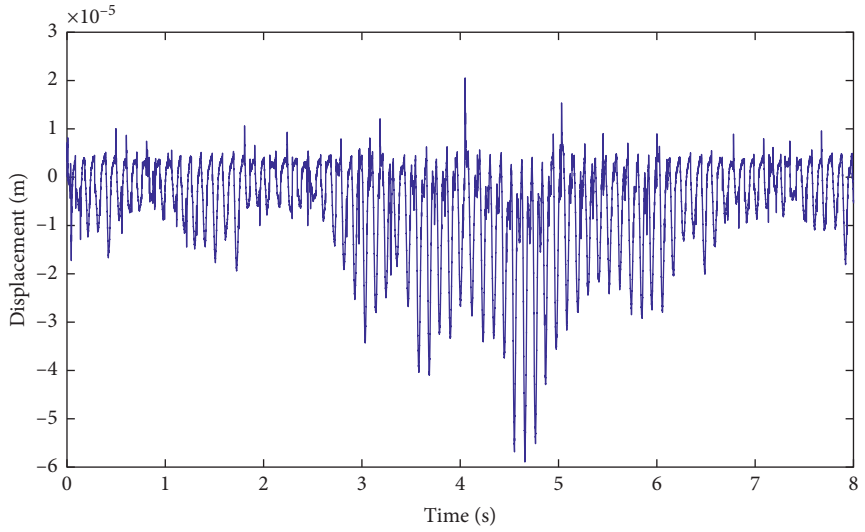
Also, the maximum amount of displacement in terms of land is shown in Table 4. As can be seen from this table, the



(a)

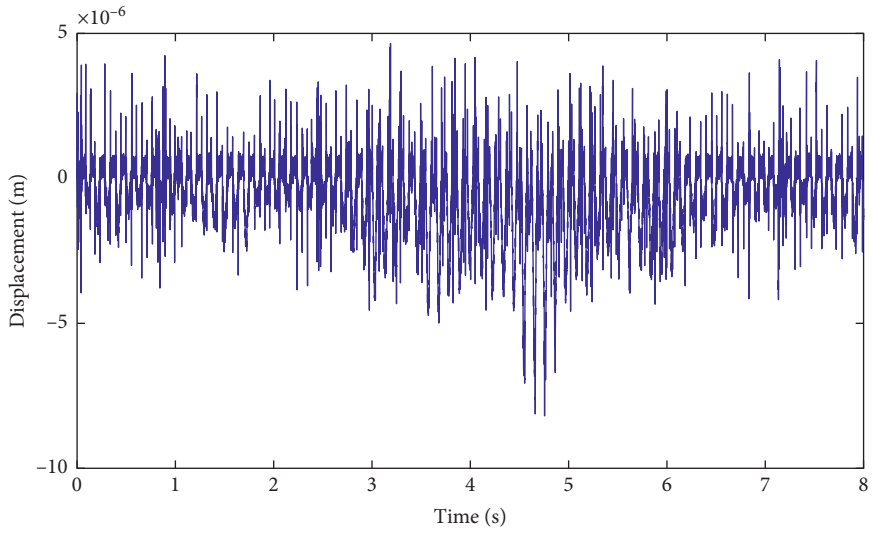


(b)



(c)

Figure 9: Continued.

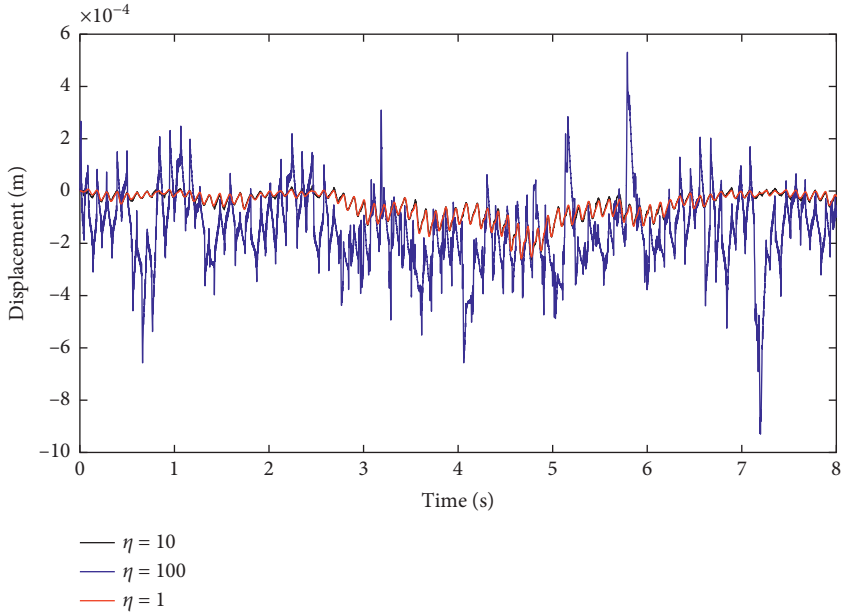


(d)

Figure 9: Controlled system behavior for different magnitudes of λ and $\eta = 10$. (a)$\lambda = 1$, (b). $\lambda = 5$, (c). $\lambda = 10$, and (d). $\lambda = 100$

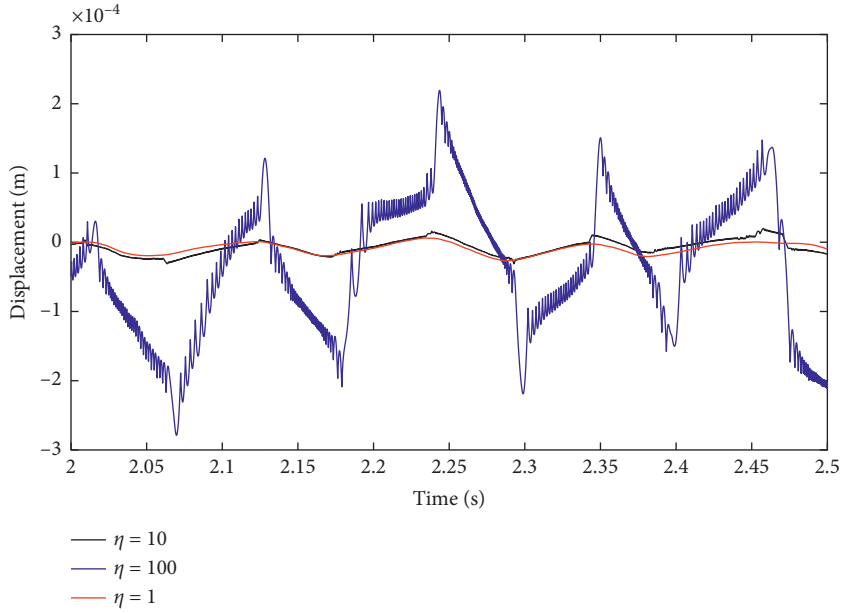
Table 4: Maximum displacement versus λ.

λ	1	2	5	10	100
Maximum displacement	0.000263	0.000167	9.4×10^{-5}	5.89×10^{-5}	8.19×10^{-6}



(a)

Figure 10: Continued.

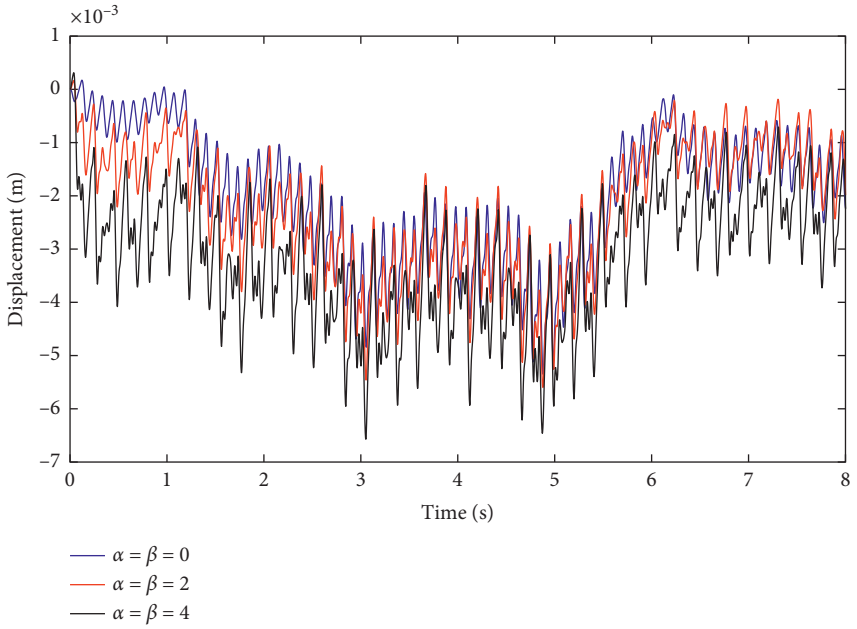


(b)

Figure 10: Controlled system behavior for different magnitudes of η and $\lambda = 1$. (a) (b). Detailed view of2 (a).

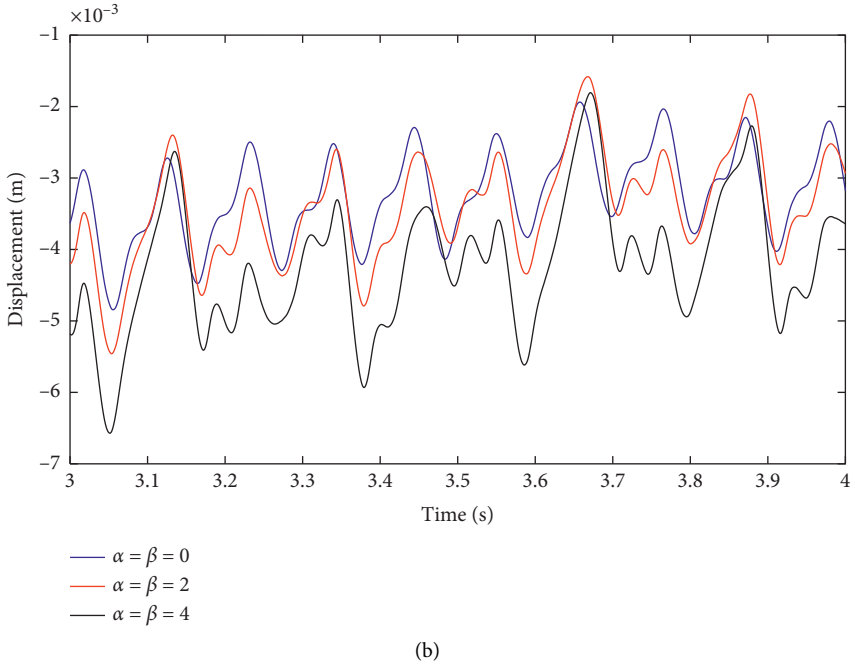
Table 5: Maximum displacement versus η.

η	1	10	100
Maximum displacement	0.000258	0.000263	0.000923



(a)

Figure 11: Continued.



(b)

Figure 11: Controlled system behavior in the presence of system uncertainties. (a), (b). Detailed view of (a).

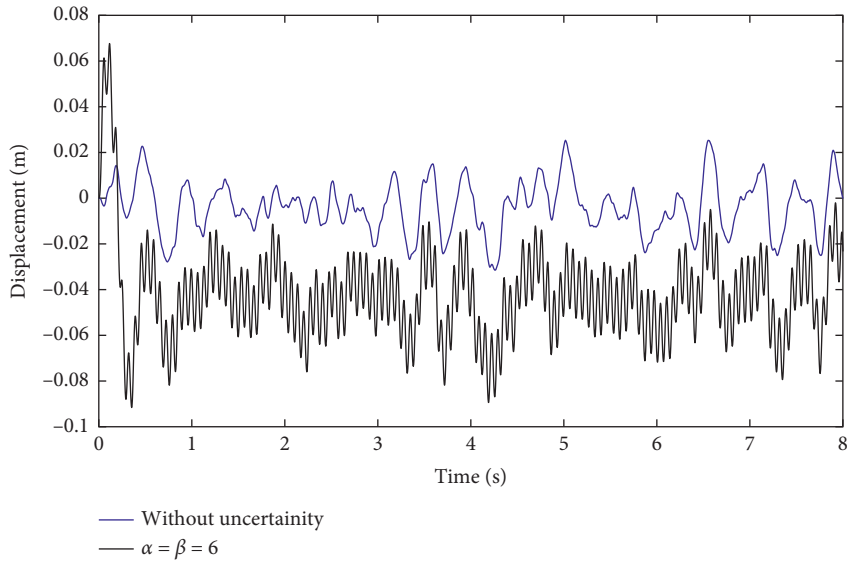


Figure 12: Uncontrolled system behavior in the presence of system uncertainties

maximum amount of displacement decreases with the increase of land increases.

The behavior of the controlled system for various values of η and $\lambda = 1$ is also shown in Figure 10. As evident in the figure, the amplitude of the system for $\eta = 100$ is considerably higher than that at $\eta = 1$ and $\eta = 10$.

The maximum displacement in terms of different η is also shown in Table 5. As can be seen from this table, the maximum amount of displacement increases with the increase of η. But as can be seen from this table, the maximum displacements for $\eta = 1$ and $\eta = 10$ are almost the same, but the maximum displacement for $\eta = 100$ is significant.

In this part, we intend to examine controller robustness in the presence of system uncertainties. The results are obtained for $\sigma = 0.05$, $\lambda = 1$, and $\eta = 10$, where $f_1(x,t) = \alpha \times 10^3 \sin t$ and $f_2(x,t) = \beta \times 10^3 \sin(t)$. The behavior of the controlled system for different α and β values is demonstrated in Figure 11. As shown by these figures, the sliding mode controller is highly capable of reducing the effects of uncertainty.

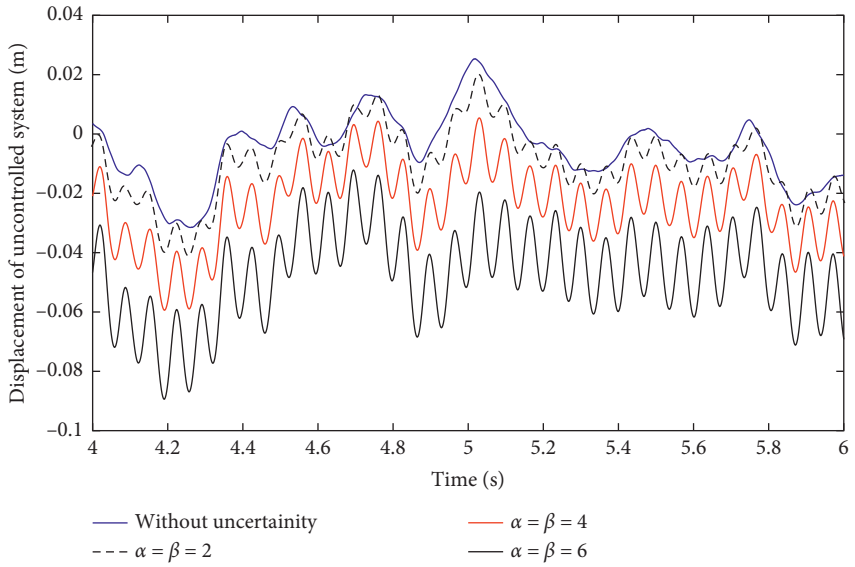
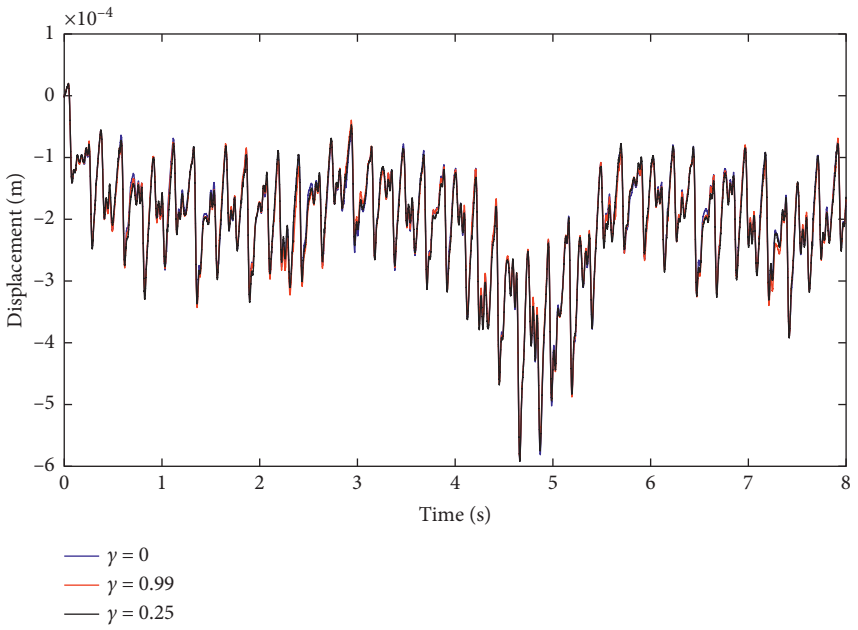


Figure 13: Uncontrolled system behavior in the presence of system uncertainties (detailed view).

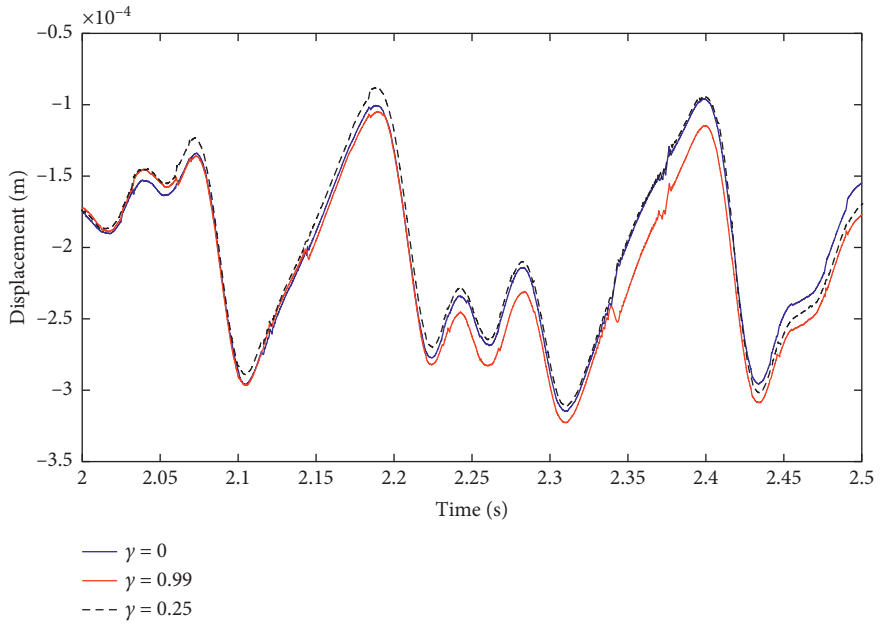
Table 6: Maximum displacement for various amounts of α and β.

$\alpha = \beta$	0	2	4	6
Maximum displacement	0.00535	0.005599	0.006572	0.09158



(a)

Figure 14: Continued.



(b)

Figure 14: Controlled system behavior for different values of γ and $\alpha = \beta = 3$. (a), (b). Detailed view of Figure 13(a).

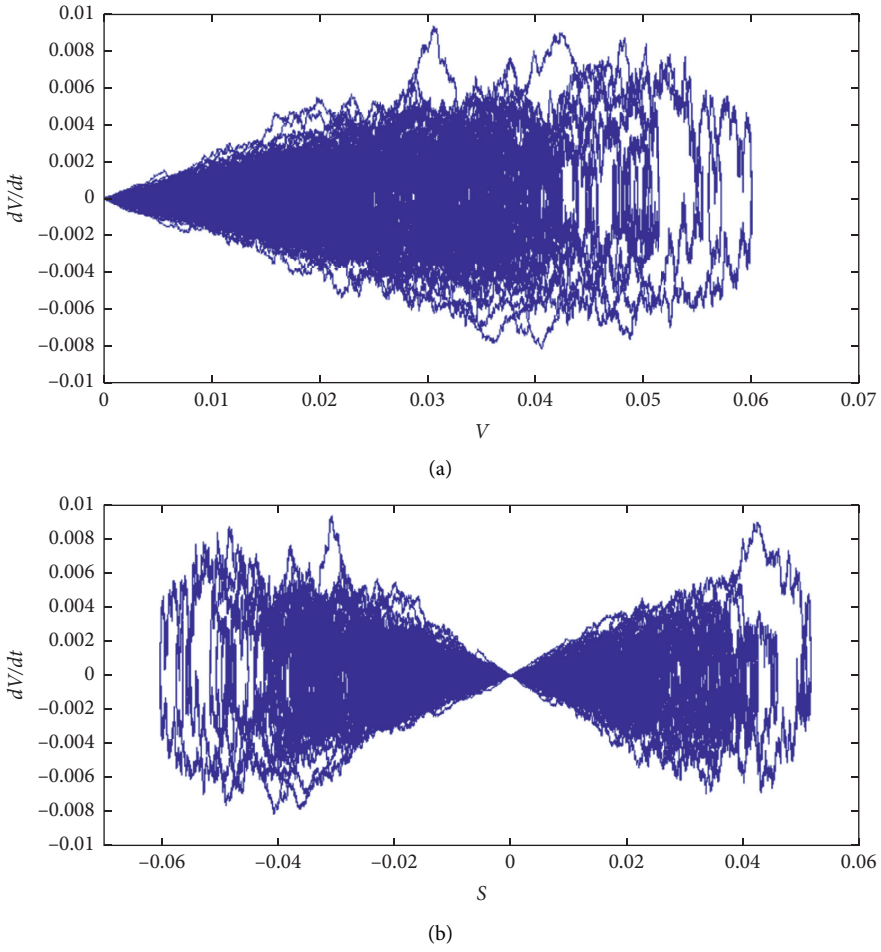


Figure 15: Variation of dV/dt versus V and s for the uncontrolled system. (a). dV/dt versus V and (b). dV/dt versus s.

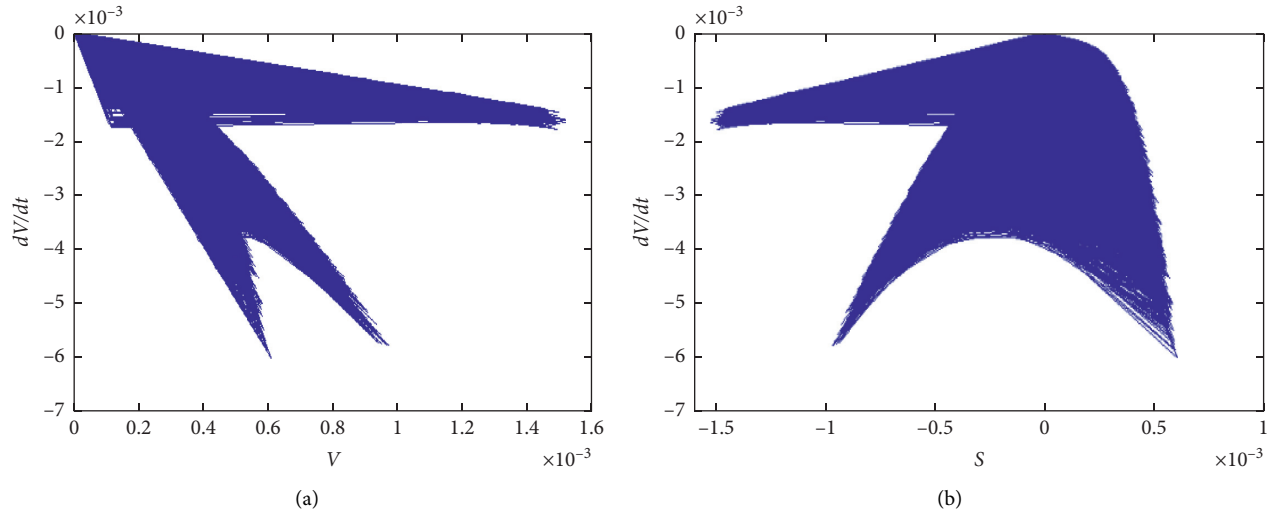


Figure 16: dV/dt versus V and s for the controlled system with $\eta = 1$. (a). dV/dt versus V and (b). dV/dt versus s.

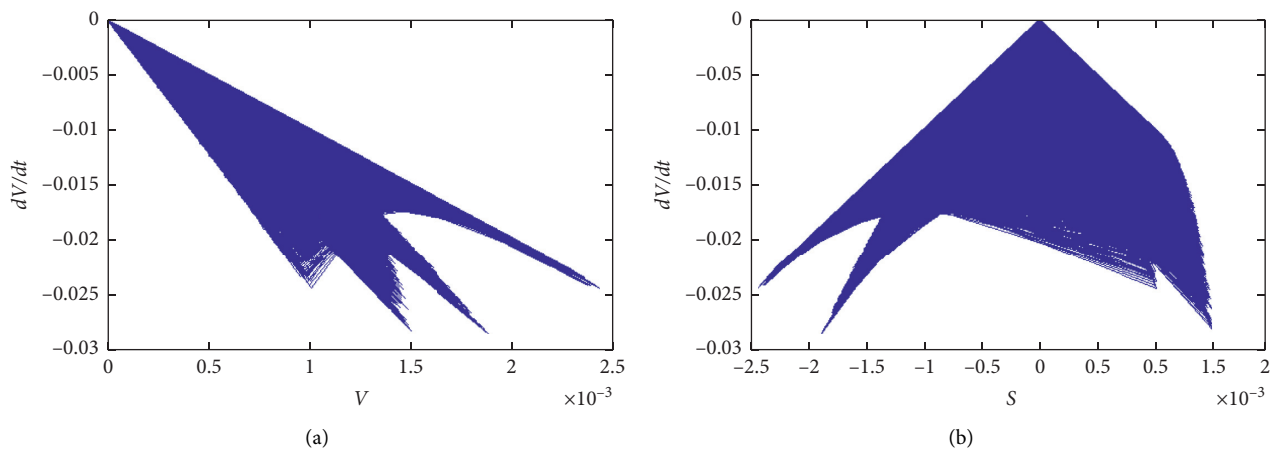


Figure 17: dV/dt versus V and s for the controlled system with $\eta = 10$. (a). dV/dt versus V. (b). dV/dt versus s.

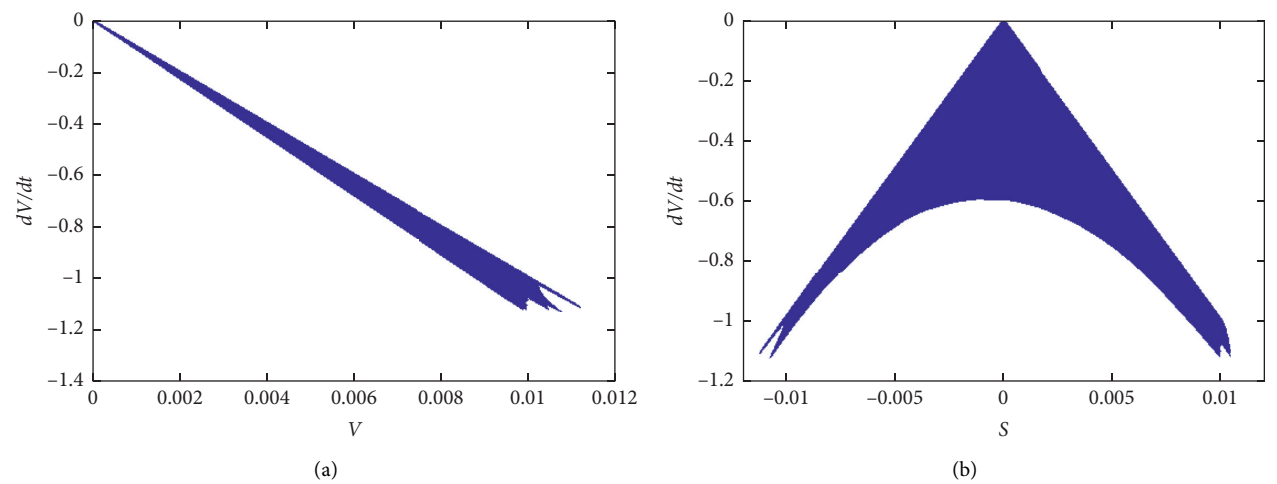


Figure 18: dV/dt versus V and s for the controlled system with $\eta = 100$. (a). dV/dt versus V, (b). dV/dt versus s.

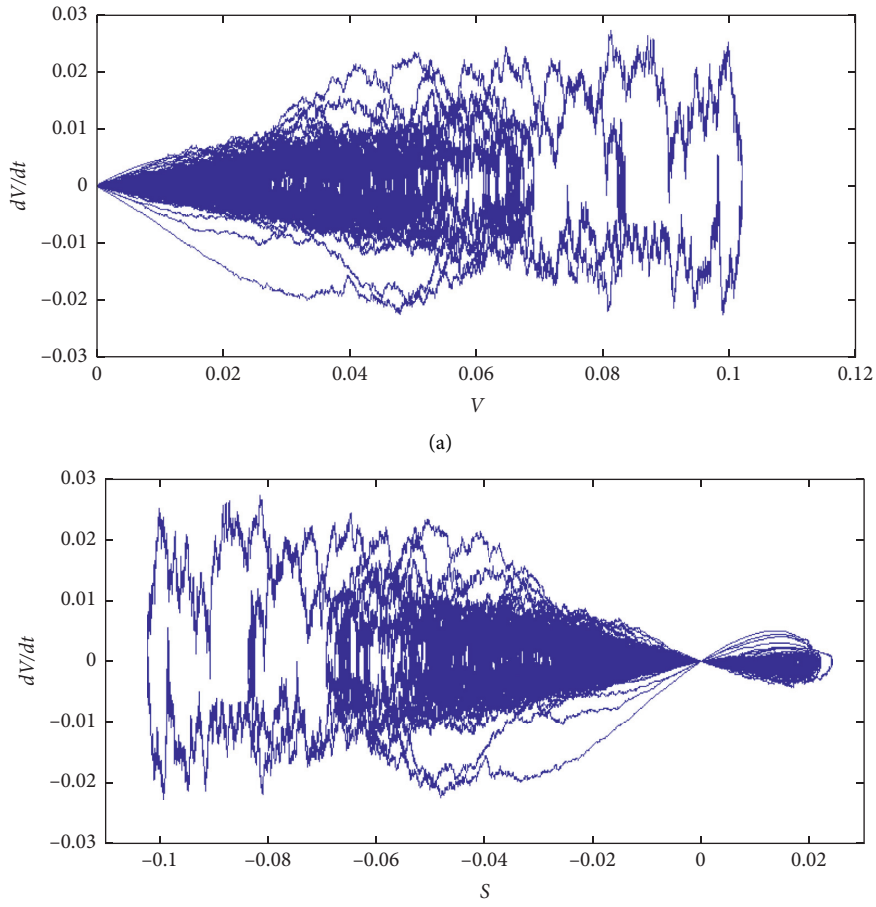


Figure 19: dV/dt versus V and s for the controlled system with $\sigma = 0.5$. (a) dV/dt versus V and (b) dV/dt versus s.

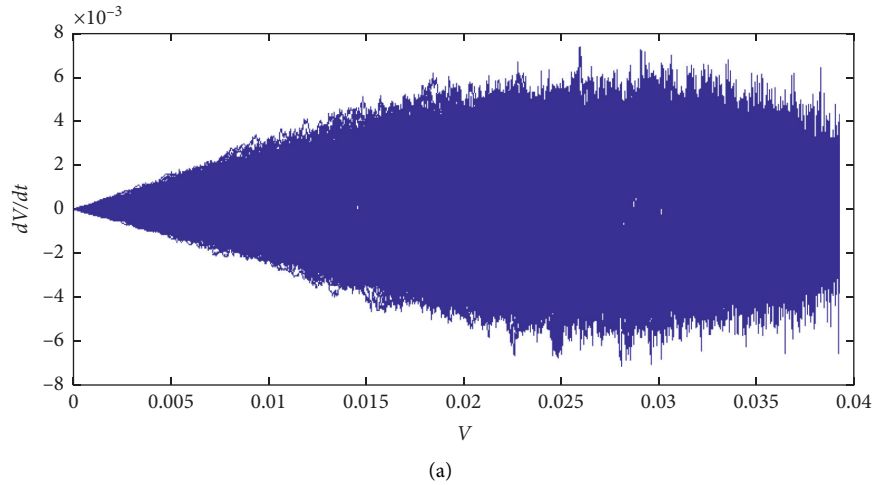
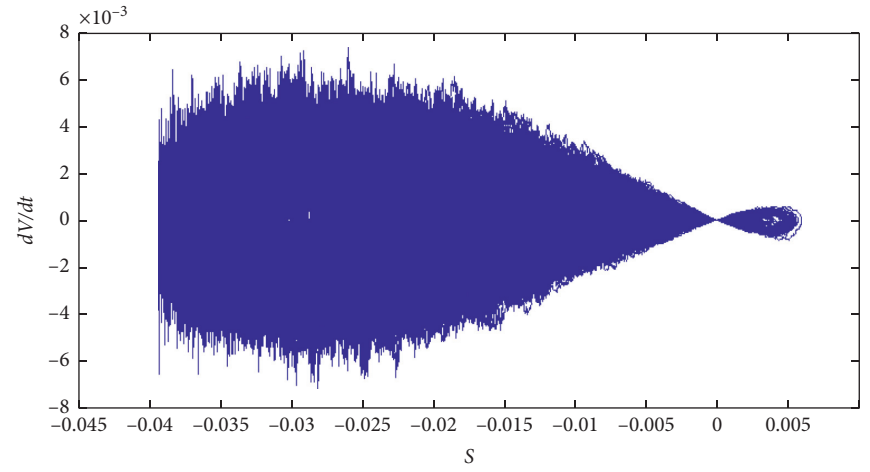
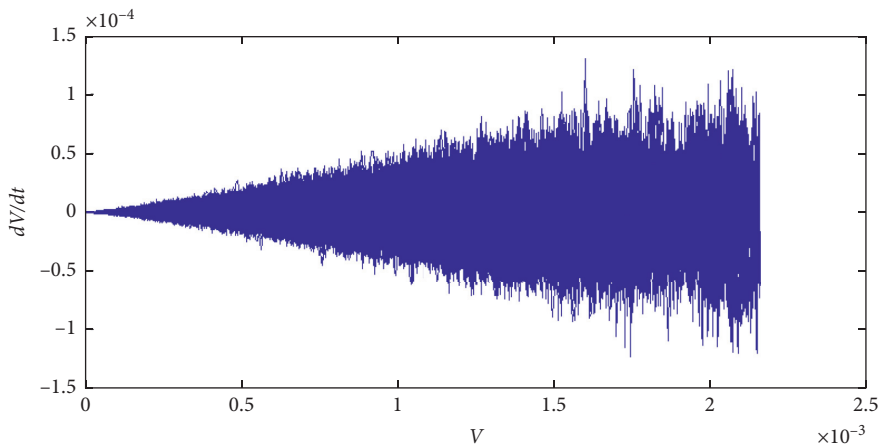


Figure 20: Continued.

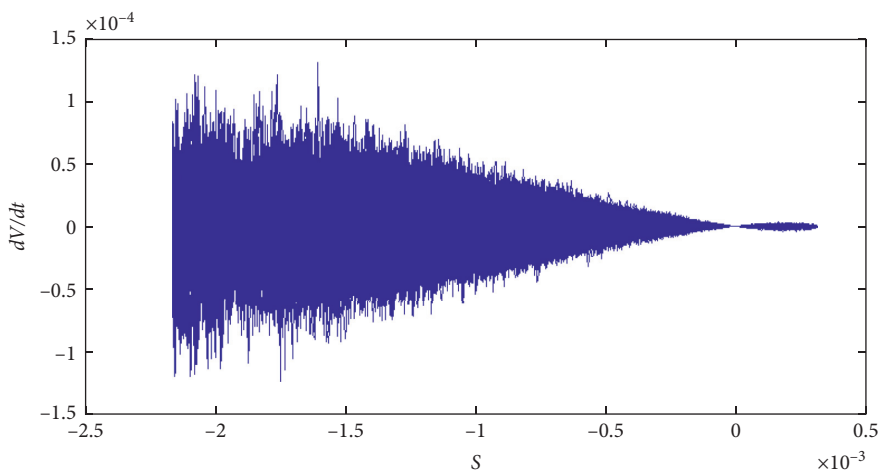


(b)

Figure 20: dV/dt versus V and s for the controlled system with $\sigma = 0.05$. (a) dV/dt versus V and (b) dV/dt versus s.



(a)



(b)

Figure 21: dV/dt versus V and s for the controlled system with $\sigma = 0.005$. (a) dV/dt versus V and (b) dV/dt versus s.

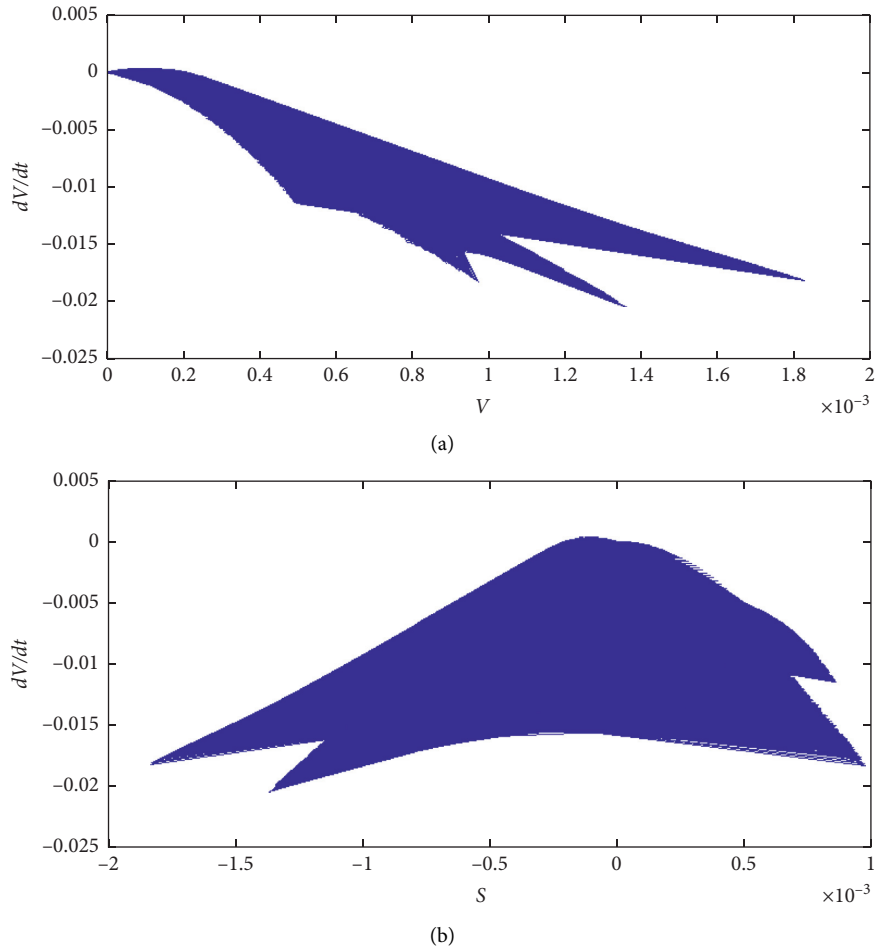


FIGURE 22: dV/dt versus V and s for the controlled system with $\sigma = 0.0005$. (a) dV/dt versus V and (b) dV/dt versus s.

The behavior of the system for different values of α and β in the passive case is shown in Figures 12 and 13. As shown, uncertainty significantly affects the behavior of the passive system, considerably increasing the amplitude of the system behavior.

The maximum displacement values shown in Figures 11 and 12 in terms of different α and β are presented in Table 6. As can be seen from this table, with increasing α and β, the maximum amount of displacement also increases. It is noteworthy that the maximum amount of displacement for the $\alpha = \beta = 6$ is significant compared to other values.

In what follows, we shall discuss the controller robustness in the face of controller uncertainty and system uncertainties. For this case, it was assumed that $\alpha = \beta = 3$ and $h(x,t) = 1 + \gamma \sin t$. The resulting system behavior is depicted in Figure 14 for different values of γ. As indicated in this figure, the controller offers high robustness in the presence of the actuator uncertainty, such that the system behavior is negligibly affected in both the absence and presence of uncertainty in the actuator. The results indicate the high robustness of the controller against actuator uncertainty.

4.2. Effect of Design Parameters and System Specifications on the Changes in the Lyapunov Function and Its Derivative. In this section, the Lyapunov function and its dependence on parameters η and σ are discussed. In Figure 15(a), the diagram of dV/dt is plotted with respect to V for the passive case. As shown, the diagram V always assumes positive values, but its derivative can take both positive and negative values. Note that $V = (1/2)s^2$, or in other words $s = \pm\sqrt{2V}$. The variation of dV/dt with respect to s was used in this paper for a more accurate analysis of the results. In Figure 15(b), the diagram of dV/dt is plotted with respect to s , which can assume both positive and negative values.

Figures 16(a), 17(a), and 18(a) depict dV/dt in terms of V for η values of 1, 10, and 100. As shown, the Lyapunov function and its derivative always assume positive and negative values, respectively. By comparing the above-mentioned figures, it follows that, as η increases, the domain or the maximum value of the Lyapunov function and its derivative grow wider. To better illustrate this, we are going to examine this observation also on the diagram of dV/dt with respect to s. Figures 16(b), 17(b), and 18(b) depict dV/dt with respect to s for η values of 1, 10, and 100. As shown, as η increases, the range of variation in $s$also increases.

Figures 17(a) and 17(b) show the diagrams dV/dt with respect to V and s. As can be seen, the range of variations in s is greater for $\eta = 10$ compared to $\eta = 1$. The diagrams dV/dt

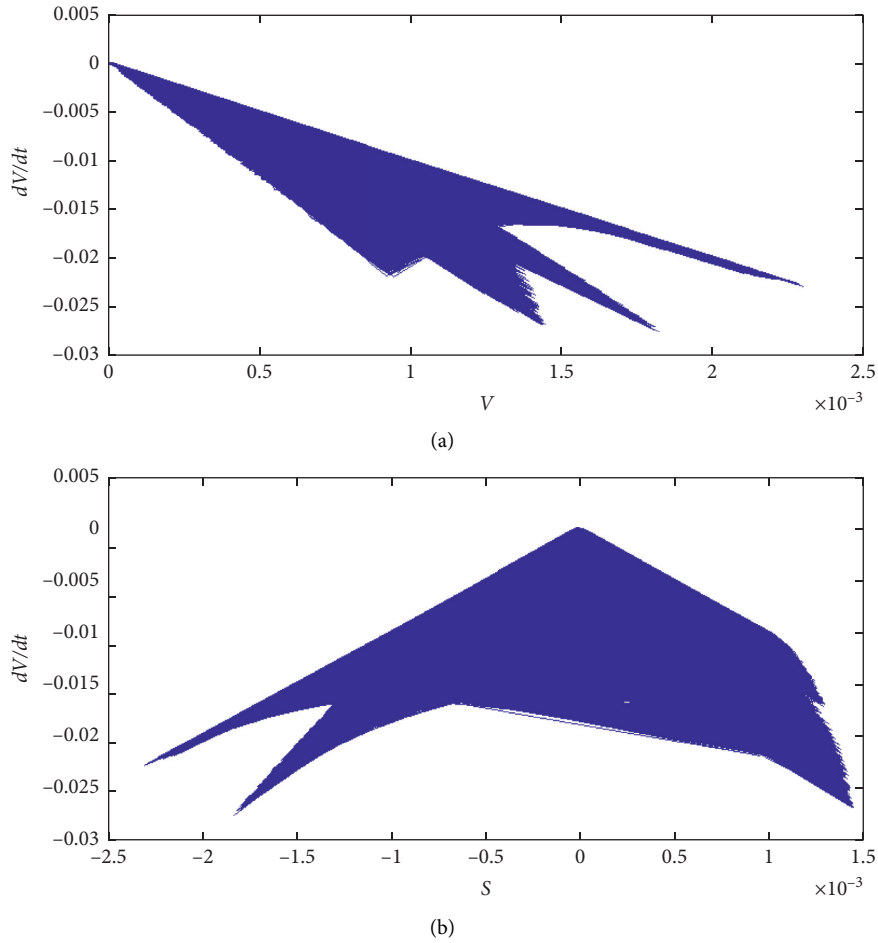


FIGURE 23: dV/dt versus V and s for the controlled system with $\sigma = 0.00005$. (a) dV/dt versus V and (b) dV/dt versus s.

with respect to V and s for $\eta = 100$ are plotted in Figures 18(a) and 18(b). As mentioned before, increasing η increases e_1 or, in other words, according to the definition of the stability of the Lyapunov, by decreasing η, the value of $\delta(\varepsilon)$ decreases, causing the attraction set to shrink, in which case the system becomes more stable according to the definition of Lyapunov stability.

In the following section, we plan to discuss changes in dV/dt with respect to V and s to eliminate chattering. As mentioned earlier, in order to eliminate the chattering phenomenon, $s/(|s| + \sigma)$ is used instead of sign(s), where σ is a small number. Note that $s/(|s| + \sigma)$is an approximation of the function sign(s), the accuracy of which is acceptable if σ is sufficiently small. However, if inappropriate values are selected for σ, the sign of dV/dt is affected, causing it to contradict its positive definite condition. In Figures 19–23, the diagram dV/dt with respect to V and s is plotted for different values of σ, $\lambda = 1$ and $\eta = 10$. A shown in Figures 19–21, dV/dt can also take positive values, meaning that the system can become temporarily unstable from the viewpoint of the Lyapunov theory. In fact, in such conditions, the chattering phenomenon is eliminated. The diagram dV/dt with respect to s for $\sigma = 0.05$ is plotted in Figure 20. As shown, the sign of the diagram dV/dt with respect to s remains unchanged in different time steps. However, as shown in Figures 22 and 23, for $\sigma = 0.0005$ and $\sigma = 0.00005$, the diagram dV/dt satisfies the absolute stability condition and dV/dt assumes negative values for all s values. As previously shown, the system exhibits a chattering behavior for $\sigma = 0.0005$ and $\sigma = 0.00005$, since dV/dt changes signs in each time step.

5. Conclusion

Given the importance of active control for car suspension systems, this paper addressed the design and dynamic analysis of a quarter-car system. To achieve more precise modeling of the suspension system, nonlinear factors in the springs and shock absorber were taken into consideration. Furthermore, uncertainty in the system and the actuator was taken into account. Road surface variations were modeled as Gaussian white noise. The sliding mode controller was employed to control the quarter-car system given that this controller is robust to uncertainty and noise and is a suitable controller for nonlinear models. The dynamic equations governing the behavior of the system were presented and the sliding mode controller was designed for the studied system. The control results for different cases of uncertainty were obtained and

compared with the behavior of a passive system. The effect of controller parameters on the system behavior and the controller robustness to uncertainties and white noise were presented. The results show that the sliding mode controller is capable of reducing uncertainty and noise effects as well as stabilizing the system. Finally, the effects of controller parameters on the system stability from Lyapunov's view as well as on the basin attraction set were presented.

After presenting of concluding remarks, the authors are interested in presenting some future works that can be studied with the presented method of this paper.

Performing such control activity is possible in the presence of white Gaussian noise for other dynamic structures, for example, investigation of the dynamic behavior of the tall building in the presence of ground excitation or earthquake, in which the ground displacement is considered as white Gaussian noise; vibrations analysis of marine structures in the presence of storms or sea waves using the active displacement reduction system, for such research, sea waves considered as white noise; aircraft wing vibrations in the presence of wind force and reducing such vibrations by using mass or vibroimpact dampers, in which wind force can be considered as Gaussian white noise; and vibration evaluation of the dynamical system in the presence of time-delayed Markov jump.

Data Availability

The related data are presented within the manuscript.

Conflicts of Interest

The authors declare they have no conflicts of interest.

Acknowledgments

This study was funded by The Research Council of Oman under Grant no. ORG/CBS/14/008.

References

[1] E. M. Papoutsis-Kiachagias, S. Porziani, C. Groth, M. E. Biancolini, E. Costa, and K. C. Giannakoglou, "Aerodynamic optimization of car shapes using the continuous adjoint method and an RBF morpher," *Computational Methods in Applied Sciences*, Springer, Cham, Switzerland, pp. 173–187, 2019.

[2] S. Thomas and O. Carsten, "Adjoint optimization for vehicle external aerodynamics," *International Journal of Automotive Engineering*, vol. 7, no. 1, pp. 1–7, 2016.

[3] R. Boodanur, A. Panwar, S. K. Kulkarni, and A. B. Jadhav, "Air intake system optimization for passenger car engine," *SAE Technical Paper*, 2019.

[4] Z. Xu, X. Xia, S. Lai, and Z. He, "Improvement of interior sound quality for passenger car based on optimization of sound pressure distribution in low frequency," *Applied Acoustics*, vol. 130, pp. 43–51, 2018.

[5] T. A. Razak, M. N. Hamid, A. M. Ghazali, S. N. Soid, and K. S. Shafee, *A Stress Analysis and Design Improvement of a Car Door Hinge for Compact Cars. InEngineering Applications for New Materials and Technologies*, pp. 151–161, Springer, Cham, Switzerland, 2018.

[6] J. Rios-Torres, J. Liu, and A. Khattak, "Fuel consumption for various driving styles in conventional and hybrid electric vehicles: integrating driving cycle predictions with fuel consumption optimization," *International Journal of Sustainable Transportation*, vol. 13, no. 2, pp. 123–137, 2019.

[7] M. Ehsani, Y. Gao, S. Longo, and K. Ebrahimi, *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles*, CRC Press, Boca Raton, FL, USA, 2018.

[8] B. S. Kong and D. K. Park, "Design optimization of the cowl cross bar - light cowl cross bar satisfying 5 performances -," *International Journal of Automotive Technology*, vol. 19, no. 3, pp. 387–391, 2018.

[9] K. Suh and H. Yoon, "Design optimization of a rear independent suspension for the Korean light tactical vehicle," *International Journal of Automotive Technology*, vol. 19, no. 2, pp. 245–252, 2018.

[10] H. Pan and W. Sun, "Nonlinear output feedback finite-time control for vehicle active suspension systems," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2073–2082, 2018.

[11] H. Pan, X. Jing, W. Sun, and H. Gao, "A bioinspired dynamics-based adaptive tracking control for nonlinear suspension systems," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 3, pp. 903–914, 2017.

[12] H. Pan, H. Li, W. Sun, and Z. Wang, "Adaptive fault-tolerant compensation control and its application to nonlinear suspension systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 5, pp. 1766–1776, 2018.

[13] M. Nagarkar, Y. Bhalerao, G. V. Patil, and R. Z. Patil, "Multi-objective optimization of nonlinear quarter car suspension system - PID and LQR control," *Procedia Manufacturing*, vol. 20, pp. 420–427, 2018.

[14] Y. Qin, C. Xiang, Z. Wang, and M. Dong, "Road excitation classification for semi-active suspension system based on system response," *Journal of Vibration and Control*, vol. 24, no. 13, pp. 2732–2748, 2018.

[15] Y. Xia, M. Fu, C. Li, F. Pu, and Y. Xu, "Active disturbance rejection control for active suspension system of tracked vehicles with gun," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4051–4060, 2017.

[16] S. M. H. Rizvi, M. Abid, A. Q. Khan, S. G. Satti, and J. Latif, "H∞control of 8 degrees of freedom vehicle active suspension system," *Journal of King Saud University - Engineering Sciences*, vol. 30, no. 2, pp. 161–169, 2018.

[17] H. P. Wang, G. I. Y. Mustafa, and Y. Tian, "Model-free fractional-order sliding mode control for an active vehicle suspension system," *Advances in Engineering Software*, vol. 115, pp. 452–461, 2018.

[18] A. Mardani, "Energy harvesting, handling and ride comfort trade-off between passive and active suspension systems of half vehicle model using PID controller for off-road vehicles," *Journal of Advances in Vehicle Engineering*, vol. 3, pp. 150–160, 2017.

[19] H. Khodadadi and H. Ghadiri, "Self-tuning PID controller design using fuzzy logic for half car active suspension system," *International Journal of Dynamics and Control*, vol. 6, no. 1, pp. 224–232, 2018.

[20] Y. Jiang, S. Yin, J. Dong, and O. Kaynak, "A review on soft sensors for monitoring, control and optimization of industrial processes," *IEEE Sensors Journal*, 2020.

[21] Y. Jiang and S. Yin, "Recursive total principle component regression based fault detection and its application to vehicular cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1415–1423, 2017.

[22] Y. Wang, Y. Xia, H. Li, and P. Zhou, "A new integral sliding mode design method for nonlinear stochastic systems," *Automatica*, vol. 90, pp. 304–309, 2018.

[23] Y. Wang, X. Xie, M. Chadli, S. Xie, and Y. Peng, "Sliding mode control of fuzzy singularly perturbed descriptor systems," *IEEE Transactions on Fuzzy Systems*, p. 1, 2020.

[24] H. Mobki, A. M. Sabegh, A. Azizi, and H. M. Ouakad, "On the implementation of adaptive sliding mode robust controller in the stabilization of electrically actuated micro-tunable capacitor," *Microsystem Technologies*, vol. 26, no. 12, pp. 3903–3916, 2020.

[25] M. P. Aghababa, "A fractional sliding mode for finite-time control scheme with application to stabilization of electrostatic and electromechanical transducers," *Applied Mathematical Modelling*, vol. 39, no. 20, pp. 6103–6113, 2015.

[26] M. P. Aghababa, "Synchronization and stabilization of fractional second-order nonlinear complex systems," *Nonlinear Dynamics*, vol. 80, no. 4, pp. 1731–1744, 2015.

[27] S. Mirzajani, M. P. Aghababa, and A. Heydari, "Adaptive T-S fuzzy control design for fractional-order systems with parametric uncertainty and input constraint," *Fuzzy Sets and Systems*, vol. 365, pp. 22–39, 2019.

[28] M. P. Aghababa and S. Moradi, "Robust adaptive dynamic surface back-stepping tracking control of high-order strict-feedback nonlinear systems via disturbance observer approach," *International Journal of Control*, vol. 4, pp. 1–17, 2020.

[29] A. M. Shotorbani, A. Ajami, S. G. Zadeh, M. P. Aghababa, and B. Mahboubi, "Robust terminal sliding mode power flow controller using unified power flow controller with adaptive observer and local measurement," *IET Generation, Transmission & Distribution*, vol. 8, no. 10, pp. 1712–1723, 2014.

[30] H. Zhang, X. Liu, J. Wang, and H. R. Karimi, "Robust H∞ sliding mode control with pole placement for a fluid power electrohydraulic actuator (EHA) system," *The International Journal of Advanced Manufacturing Technology*, vol. 73, no. 5-8, pp. 1095–1104, 2014.

[31] A. Azizi, "Computer-based analysis of the stochastic stability of mechanical structures driven by white and colored noise," *Sustainability*, vol. 10, no. 10, p. 3419, 2018.

[32] A. Azizi, "A case study on computer-based analysis of the stochastic stability of mechanical structures driven by white and colored noise: utilizing artificial intelligence techniques to design an effective active suspension system," *Complexity*, vol. 25, p. 8, 2020.

[33] A. Azizi and P. Ghafoorpoor Yazdi, "Modeling and control of the effect of the noise on the mechanical structures," in *Computer-Based Analysis of the Stochastic Stability of Mechanical Structures Driven by White and Colored Noise*, pp. 75–93, Springer, Singapore, 2019.

[34] A. Azizi and P. Ghafoorpoor Yazdi, "Noise control techniques," in *Computer-Based Analysis of the Stochastic Stability of Mechanical Structures Driven by White and Colored Noise*, pp. 61–73, Springer, Singapore, 2019.

[35] M. P. Aghababa, "Comments on "Fuzzy fractional order sliding mode controller for nonlinear systems," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 3, pp. 1489–1492, 2012.

[36] H. Yang, Y. Jiang, and S. Yin, "Fault-tolerant control of time-delay Markov jump systems with $It\hat{o}$ stochastic process and output disturbance based on sliding mode observer," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 12, pp. 5299–5307, 2018.

[37] M. F. Ismail, K. Peng, N. Hamzah, Y. M. Sam, M. K. Aripin, and M. H. Hasan, "A linear model of quarter car active suspension system using composite nonlinear feedback control," in *Proceedings of the 2012 IEEE Student Conference on Research and Development (SCOReD)*, pp. 98–103, Pulau Pinang, Malaysia, 2012.

[38] N. Ishak, R. S. Othman, A. Ahmad, Y. M. Sam, and A. A. Basari, "An observer design of nonlinear quarter car model for active suspension system by using backstepping controller," in *Proceedings of the 2009 5th International Colloquium on Signal Processing & Its Applications*, pp. 160–165, Kuala Lumpur, Malaysia, 2009.

[39] H. Mobki, M. Jalilirad, M. V. Moradi, and A. Azizi, "Multi input versus single input sliding mode for closed-loop control of capacitive micro structures," *SN Applied Sciences*, vol. 1, no. 7, pp. 1–13, 2019.

[40] D. J. Higham., "An algorithmic introduction to numerical simulation of stochastic differential equations," *SIAM Review*, vol. 43, no. 3, pp. 525–546, 2001.

[41] A. Ajami, A. M. Shotorbani, and M. P. Aagababa, "Application of the direct Lyapunov method for robust finite-time power flow control with a unified power flow controller," *IET Generation, Transmission & Distribution*, vol. 6, no. 9, pp. 822–830, 2012.

[42] M. P. Aghababa, "Lyapunov control method for mismatched uncertainty and gain variation compensation in switched systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–11, 2020.

[43] M. P. Aghababa and H. Feizi, "Nonsingular terminal sliding mode approach applied to synchronize chaotic systems with unknown parameters and nonlinear inputs," *Chinese Physics B*, vol. 21, no. 6, Article ID 060506, 2012.

[44] A. Medio and M. Lines, *Nonlinear Dynamics: A Primer*, Cambridge University Press, Cambridge, UK, 2001.

[45] J. J. Slotine and W. Li, *Applied Nonlinear Control*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1991.

[46] H. Salarieh and A. Alasty, "Control of stochastic chaos using sliding mode method," *Journal of Computational and Applied Mathematics*, vol. 225, no. 1, pp. 135–145, 2009.

[47] A. S. Poznyak, "Stochastic sliding mode control and state estimation," in *Advances in Variable Structure Systems and Sliding Mode Control - Theory and Applications*, pp. 57–100, Springer, Cham, Switzerland, 2018.

[48] A. Poznyak, "Stochastic super-twist sliding mode controller," *IEEE Transactions on Automatic Control*, vol. 63, no. 5, pp. 1538–1544, 2018.

[49] E. R. Da Silva, E. Assunção, M. Teixeira, and L. F. Buzachero, "Less conservative control design for linear systems with polytopic uncertainties via state-derivative feedback," *Mathematical Problems in Engineering*, vol. 2012, Article ID 315049, 21 pages, 2012.

[50] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, "Linear matrix inequalities in system and control theory," *Society For Industrial And Applied Mathematics*, 1994.

[51] P. Gahinet, A. Nemirovskii, A. J. Laub, and M. Chilali, "The LMI control toolbox," in *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, vol. 3, pp. 2038–2041, Lake Buena Vista, FL, USA, 1994.

Hindawi
Complexity
Volume 2021, Article ID 6651105, 15 pages
https://doi.org/10.1155/2021/6651105

Research Article

Repetitive Control Scheme of Robotic Manipulators Based on Improved B-Spline Function

Xingyu Wang[1], **Anna Wang**[1], **Dazhi Wang**[1], **Wenhui Wang**[1], **Bingxue Liang**[2] and **Yufei Qi**[3]

[1]*College of Information Science and Engineering, Northeastern University, Shenyang 110819, China*
[2]*Bluelight Drive Technology Co., Ltd., Shenyang 110179, China*
[3]*China North Vehicle Research Institute, Beijing 100072, China*

Correspondence should be addressed to Anna Wang; wanganna@mail.neu.edu.cn

Received 10 December 2020; Revised 18 March 2021; Accepted 8 April 2021; Published 17 April 2021

Academic Editor: Aydin Azizi

In this paper, a repetitive control scheme of a 2-DOF robotic manipulator based on the improved cubic B-spline curve is proposed. Firstly, a repetitive controller for robotic manipulator is designed, which is composed of an iterative controller and disturbance observer. Then, an improved B-spline optimization scheme is introduced to divide the task of the robotic manipulator into three intervals. A correction function is added to each interval of cubic spline interpolation. Finally, a variety of cases are designed and simulated by MATLAB. The experimental results show that, compared with the conventional B-spline, the improved B-spline has better performance in tracking accuracy and smoothness of motion trajectory. By changing the mechanism of the manipulator, the cases of different weights and lengths are designed. The experimental results in these cases show that the proposed scheme can be applied to most of the 2-DOF robotic manipulator control systems.

1. Introduction

Manipulators have been commonly employed in industrial manufacturing fields, including spot welding, painting, machine operation, electronic packaging, and palletizing. The robotic manipulator system will replace the traditional manual work and become the mainstream of industrial development. The robotic manipulator control system is a multi-input, multi-output, highly coupled nonlinear complex system, which is affected by many uncertainties and interference factors. In particular, the trajectory control of robot is very complex, and it is difficult to establish a dynamic model. In recent years, a practical iterative learning control method is proposed in [1, 2]. This method has the characteristics of simple structure, small amount of calculation, and no need of accurate mathematical model of the system. It is very suitable for solving the problem of control object under repetitive working mode. In the practical application of robotics and automatic machines, the tasks performed are usually repetitive or periodic. Many of these tasks merely imply a continuous repetition of a given path. From the control point of view, it is necessary to identify the periodic external signals that can be considered as known, because they refer to planned trajectories or disturbances, and their cycle times can be easily measured or known in advance. In order to improve the trajectory tracking accuracy, repetitive control (RC) is a simple design but effective method, because it eliminates the tracking error in the repetitive process by iterative learning. RC was first developed by Inoue et al. [3, 4]. It has been applied to many other proton synchrotron power supply systems. The successful applications of RC in many applications are reported in [5, 6], such as high-precision trajectory tracking of servo mechanism, torque vibration suppression in motor, noise elimination in power supply, and industrial manipulator.

In recent years, many control techniques have been proposed for manipulators. Wang and Zhang proposed an adaptive neural network tracking control scheme for robotic manipulator, which achieved good performance in dealing with model uncertainty [7]. Aydin proposed a robust sliding

mode control scheme to suppress model uncertainty [8]. Mu et al. proposed a PD feedback control law to deal with the parameter changes when capturing objects [9]. Dai et al. proposed a model predictive control scheme for redundant manipulators with online identification of unknown parameters [10]. Xiang and Yan analyzed the uncertainty model of the dynamic model with clearance joints [11]. In [12], an inverse dynamic control method is proposed to solve the nonlinear problem of industrial robotic manipulator. Lu and Jia proposed an Adaptive Coordinated Control for a free-floating space manipulator with uncertain kinematics and dynamics characteristics [13]. However, the above schemes do not consider the repetitive reference trajectory tracking control under model uncertainty. In fact, considering that the existence of vibration interference has a great impact on repetitive motion, repetitive reference trajectory tracking plays an important role in many cases.

Industrial manipulator control has various characteristics, among which the control system with unknown parameters or uncertain model is the difficulty. Therefore, the control of complex systems such as robot manipulator system with uncertainties has become a worthy research topic. An improved funnel function feedback controller is proposed for nonlinear servo mechanism [14]. In [15], an unknown input observer and an adaptive parameter estimation method are proposed to solve the uncertainty of engine torque. Considering the uncertainty of the contact surface of the manipulator, Li et al. proposed a decentralized adaptive fuzzy control method [16]. For the manipulator system, Fang et al. proposed a fuzzy control scheme with adaptive algorithm, simplified the logic structure, and optimized the amount of calculation to adapt to the practical application [17]. Wang et al. designed a new nonsingular fast terminal sliding surface and used time delay estimation technology to eliminate the complex nonlinear effect of the manipulator [18]. In [19], adaptive control is introduced into time delay estimation to improve the accuracy and speed of nonsingular fast terminal sliding mode. In order to solve the influence of various uncertain factors in manipulator system, a variety of observers were proposed in [20] to deal with uncertain kinematics and estimate unknown torque, respectively. References [21, 22] have proposed a real-time online error compensation method for the manipulator, which avoids frequent calibration of sensors in the actual industrial application scene and reduces the complexity of the control system. By using the filter to generate uniform B-spline trajectory, the tracking error at the desired path point can be eliminated, and the trajectory error at the end of the manipulator can be compensated without modifying the core content of the controller [23]. It is especially suitable for industrial equipment that cannot modify the controller.

B-spline can provide continuity of velocity and acceleration at lower order of each knot. In some literatures, the best smooth trajectory planning method of rigid manipulator based on B-spline has been adopted. In [24], higher-order B-splines are applied to the motion planning of serial manipulators. In [25], the acceleration and velocity signal trajectories of rigid link parallel robot are identified and optimized by using quintic B-spline. In [26], spline function is used to optimize the reference input signal of servo driver, and its parameters are determined by particle swarm optimization algorithm, in order to reduce tracking error. In [23], uniform B-spline curve trajectory was generated by dynamic filter to modify reference signal in real time. Qian et al. proposed a trajectory planning method for cable driven parallel robot based on improved quintic B-spline curve [27]. The control points of these B-spline trajectory compensation methods are evenly distributed, and there is no special treatment for some demanding trajectories. In practical application, too much calculation is used in the uncomplicated trajectories, but the performance is not satisfactory in the parts with more uncertain disturbances.

In this paper, a new method of smoothing the trajectory of robotic manipulator by using cubic B-spline is proposed. It can reduce the flexible impact and vibration disturbance produced by the actuator when the robotic manipulator moves repeatedly. In addition, an improved B-spline trajectory compensation scheme is proposed in this paper. Compared with the conventional uniform B-spline trajectory [24], it has advantages in trajectory tracking accuracy and trajectory smoothness. This paper is arranged as follows. Section 2 introduces the structure of 2-DOF robotic manipulator and the dynamic model of the mechanism. In Section 3, a repetitive controller for robotic manipulator is designed. In Section 4, according to the task characteristics of the robotic manipulator, the characteristics of the interference signals in different stages are studied, and then the correction function is introduced to optimize the conventional B-spline. In Section 5, MATLAB is used to simulate the trajectory tracking accuracy of some cases. The simulation results show that the improved B-spline has better performance than the traditional B-spline. Conclusions are drawn in Section 6.

2. Model Derivation

We refer to some novel modelling methods of robotic manipulator [11], and combining with the data of this paper, we propose the following model derivation for robotic manipulator.

The dynamic equation of the N joint manipulator is

$$\mathbf{M}(q(t))\ddot{q}(t) + \mathbf{C}(q(t), \dot{q}(t))\dot{q}(t) + \mathbf{g}(q(t)) = \boldsymbol{\tau}(t). \quad (1)$$

In addition, in practical application, the manipulator will bear uncertain interference such as parameter change, external disturbance, and friction. For example, due to the production process, assembly error, friction, wear, and other reasons, there are inevitable design tolerances for the parts of the mechanical arm; due to the movement, it will cause the change of the centre of gravity, which will lead to the change of the parameters of the mechanical arm; due to the fact that the elastic deformation will occur when the mechanical arm grasps and places heavy objects, there is an unmodeled dynamic; there are a lot of external disturbances in the working process. We conclude that the external disturbance is mainly caused by pitch vibration and vertical vibration. In

this paper, the characteristics of two kinds of external disturbances are studied and the model is established.

These factors bring about difficulties to the control of the manipulator and make it unable to track the given trajectory. Therefore, the dynamic model of the actual manipulator system should include the description of various uncertainties of the system. The mathematical model is as follows:

$$\mathbf{M}(q(t))\ddot{q}(t) + \mathbf{C}(q(t), \dot{q}(t))\dot{q}(t) + \mathbf{g}(q(t)) + \mathbf{f}(\dot{q}(t)) + \boldsymbol{\tau}_d(t) = \boldsymbol{\tau}(t). \tag{2}$$

As shown in Figure 1, the dynamic model of the 2-DOF manipulator using the Lagrange method can be obtained as

$$\mathbf{M}(q(t)) = \begin{bmatrix} l_2^2 m_2 + 2l_1 l_2 m_2 \cos\theta_2 + l_1^2(m_1 + m_2) & l_2^2 m_2 + l_1 l_2 m_2 \cos\theta_2 \\ l_2^2 m_2 + l_1 l_2 m_2 \cos\theta_2 & l_2^2 m_2 \end{bmatrix}, \tag{3}$$

$$\mathbf{C}(q(t), \dot{q}(t)) = \begin{bmatrix} -m_2 l_1 l_2 \dot{\theta}_2^2 \sin\theta_2 & -2m_2 l_1 l_2 (\dot{\theta}_1 + \dot{\theta}_2) \sin\theta_2 \\ m_1 l_1 l_2 \dot{\theta}_1 \sin\theta_2 & 0 \end{bmatrix}, \tag{4}$$

$$\mathbf{g}(q(t)) = \begin{bmatrix} m_2 l_2 g \cos(\theta_1 + \theta_2) + (m_1 + m_2) l_1 g \cos\theta_1 \\ m_2 l_2 g \cos(\theta_1 + \theta_2) \end{bmatrix}. \tag{5}$$

Considering the pitching vibration, it is necessary to add the base angle θ_0 into the gravity matrix as

$$\mathbf{g}(q(t)) = \begin{bmatrix} (m_1 + 2m_2) g l_1 \cos(\theta_0 + \theta_1) + m_2 g l_2 \cos(\theta_0 + \theta_1 + \theta_2) \\ m_2 l_2 g \cos(\theta_0 + \theta_1 + \theta_2) \end{bmatrix}, \tag{6}$$

$$\mathbf{f}(\dot{q}(t)) = \begin{bmatrix} \alpha_1 \cdot \mathrm{sgn}(\dot{\theta}_1) \\ \alpha_2 \cdot \mathrm{sgn}(\dot{\theta}_2) \end{bmatrix}, \tag{7}$$

$$\boldsymbol{\tau}(t) = \begin{bmatrix} \tau_1(t) \\ \tau_2(t) \end{bmatrix}. \tag{8}$$

3. Controller Design

Based on the existing iterative learning control method [1, 2], an iterative control scheme combined with B-spline function is proposed for the repetitive trajectory tracking control of robotic manipulator. As illustrated in Figure 2, q_0 and $\dot{q}_0$ are the desired angle and angular velocity, respectively. $q_{i\text{th}}$ and $\dot{q}_{i\text{th}}$ are the practical angle and angular velocity, respectively. $e_{i\text{th}}$ and $\dot{e}_{i\text{th}}$ are the angle tracking error and angular velocity tracking error, respectively.

In Figure 2, it can be observed that the BIC is designed to reject the vibration disturbances and attenuate model uncertainties. Additionally, it is designed to further improve the accuracy and repeatability for repetitive trajectory tracking.

3.1. Disturbance Observer Design. According to the dynamic characteristics of robotic manipulator [28], the disturbance observer is constructed as

$$\begin{cases} \hat{\tau}_d = K\mathbf{M}(q)\dot{q} + \nu, \\ \dot{\nu} = -K\hat{\tau}_d + K(\mathbf{C}(q, \dot{q})q - \tau - \mathbf{M}(q)\dot{q}), \end{cases} \tag{9}$$

where $\hat{\tau}_d$ is the estimation of disturbances τ_d and ν is an intermediate variable. The disturbance estimation error is denoted as

$$\tilde{\tau}_d = \tau_d - \hat{\tau}_d, \tag{10}$$

and the error dynamic of disturbance observer can be described as

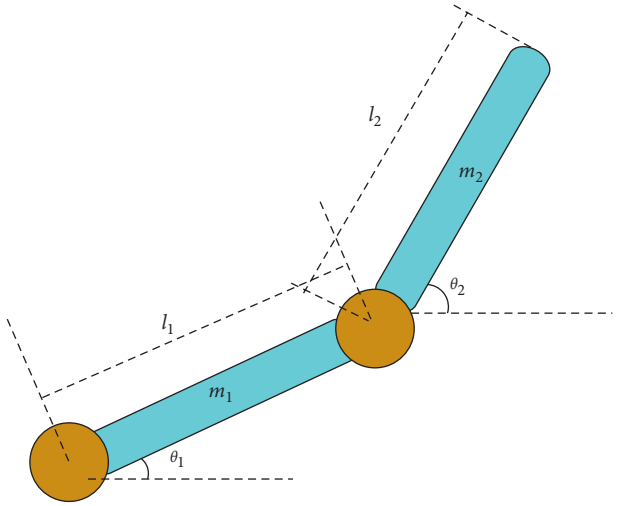


Figure 1: Simplified model of 2-DOF manipulator.

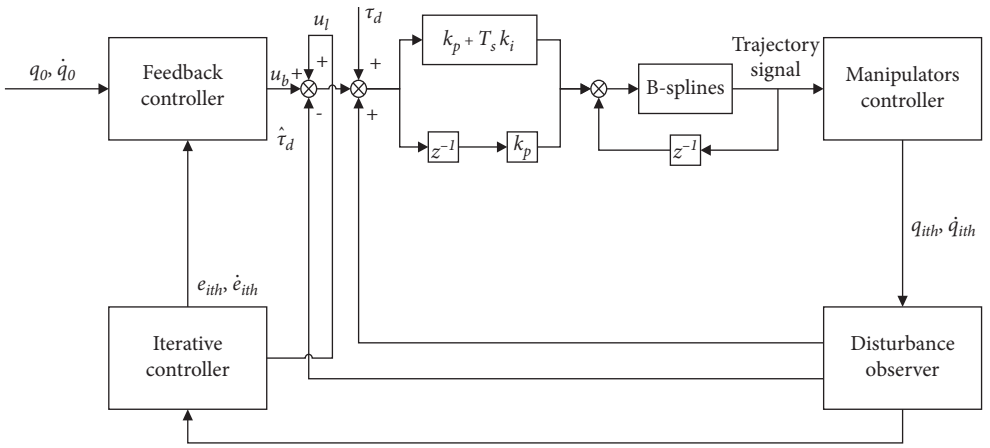


Figure 2: Configuration of iterative control scheme combined with B-spline.

$$\tau = u_b - \hat{\tau}_d, \tag{11}$$

where u_b is the feedback controller model:

$$u_b = K\mathbf{M}(q)\left(e_q - \dot{e}_q\right) + \mathbf{M}(q)\ddot{q}_0 + \mathbf{C}(q,\dot{q})\dot{q}. \tag{12}$$

Combining (10) with (2) can obtain

$$\mathbf{M}(q)\ddot{q} + \mathbf{C}(q,\dot{q})\dot{q} = K\mathbf{M}(q)\left(e_q + \dot{e}_q\right) + \mathbf{M}(q)\ddot{q}_0 + \mathbf{C}(q,\dot{q})\dot{q} - \hat{\tau}_d + \tau_d, \tag{13}$$

$$\mathbf{M}(q)\ddot{e}_q = K\mathbf{M}(q)\left(e_q + \dot{e}_q\right) + \tilde{\tau}_d. \tag{14}$$

Then, the tracking error can be written as

$$\ddot{e}_q = K\left(e_q + \dot{e}_q\right) + \mathbf{M}^{-1}(q)\tilde{\tau}_d. \tag{15}$$

Define $\varepsilon = \begin{bmatrix} e_q & \dot{e}_q \end{bmatrix}^{-1}$; then formula (15) can be written as

$$\dot{\varepsilon} = (Z + YK)\varepsilon + D\tilde{\tau}_d. \tag{16}$$

The augmented system can be written as follows:

$$\begin{bmatrix} \dot{\varepsilon} \\ \dot{\tilde{\tau}}_d \end{bmatrix} = \begin{bmatrix} Z + YK_b & D \\ 0 & K_d \end{bmatrix} \begin{bmatrix} \varepsilon \\ \tilde{\tau}_d \end{bmatrix} + \begin{bmatrix} 0 & D \\ I & -K_d \end{bmatrix} \begin{bmatrix} \dot{\tau}_d(0) \\ \tau_d(t) \end{bmatrix}. \tag{17}$$

3.2. Iterative Controller Design. The iterative controller for trajectory tracking of robotic manipulator is proposed as

$$u_{l(i+1)\text{th}} = \mathbf{M}(q)\left(u_{l(i+1)\text{th}}(t) - k_p e_q(t+\Delta) - k_d \dot{e}_q(t+\Delta)\right), \tag{18}$$

where k_p and k_d are control gains, and time shift Δ is a small positive number to be chosen. The iterative controller scheme requires $\Delta > 0$ and it can be selected small enough to specify the tracking error bounds as required, but it should be noted that the smaller Δ will slow down the convergence rate as the iteration progresses.

Combining the iterative controller with aforementioned disturbance observer and feedback controller, the control input is presented as follows:

$$\tau_i = u_b + u_{li\text{th}} - \widehat{\tau}_d. \tag{19}$$

Substituting (19) into (2), (17) can be rewritten as

$$\begin{bmatrix} \dot{\varepsilon} \\ \dot{\tilde{\tau}}_d \end{bmatrix} = \begin{bmatrix} Z + \mathrm{YK}_b & D \\ 0 & K_d \end{bmatrix} \begin{bmatrix} \varepsilon \\ \tilde{\tau}_d \end{bmatrix} + \begin{bmatrix} 0 & D \\ I & -K_d \end{bmatrix} \begin{bmatrix} \dot{\tau}_d(0) \\ \tau_d(t) \end{bmatrix} + \begin{bmatrix} D \\ 0 \end{bmatrix} u_{li\text{th}}. \tag{20}$$

The conventional iterative learning control scheme can only deal with repeated disturbances. The repetitive control scheme proposed in this paper is combined with disturbance observer, which can simultaneously control repetitive disturbance and uncertain dynamic disturbance.

4. Stability Analysis

Considering the whole robotic manipulator system full of 2 parts, considering the stability of the whole closed-loop control manipulator system, the stability proof part should also include 2 parts:

$$V = V_1 + V_2 = \varepsilon^T P_1 \varepsilon + \tilde{\tau}_d^T P_2 \tilde{\tau}_d. \tag{21}$$

Computing the derivate of V_1 andV_2 along the trajectories of (2), we can obtain

$$\begin{aligned} \dot{V}_1 &= \dot{\varepsilon}^T P_1 \varepsilon + \varepsilon^T P_1 \dot{\varepsilon} \\ &= \varepsilon^T \left(P_1 Z + P_1 \mathrm{YK}_b\right)\varepsilon + 2\varepsilon^T P_1 D\tilde{\tau}_d(0) + 2\varepsilon^T P_1 D\tau_d(t) \\ &\le \varepsilon^T \left(P_1 Z + P_1 \mathrm{YK}_b + \gamma^2 P_1 P_1^T\right)\varepsilon + \gamma^{-1}\|D\|^2 \left\|\tau_d(0)\right\|^2 + 2\varepsilon^T P_1 D\tilde{\tau}_d(0), \end{aligned} \tag{22}$$

$$\begin{aligned} \dot{V}_2 &= \dot{\tilde{\tau}}_d^T P_2 \tilde{\tau}_d + \tilde{\tau}_d^T P_2 \dot{\tilde{\tau}}_d \\ &= \left(\dot{\tau}_d(0) - K_d\tilde{\tau}_0(0) - K_d\tau(t)\right)^T P_2 \tilde{\tau}_d(0) \\ &\quad + \tilde{\tau}_d^T P_2 \left(\dot{\tau}_d(0) - K_d\tilde{\tau}(0) - K_d\tau(t)\right) \\ &= \tilde{\tau}_d^T P_2 \dot{\tau}_d - \tilde{\tau}_d^T K_d P_2 \tilde{\tau}_d - \tilde{\tau}_d^T P_2 K_d \tau_d(t) \\ &\le \tilde{\tau}_d^T \left(K_d P_2 + \gamma^2 \left(K_d P_2\right)\left(K_d P\right)^T + \gamma^2 P_2 P_2^T\right)\tilde{\tau}_d + \gamma^{-2}\tau_d^T \tau_d + \gamma^{-2}\dot{\tau}_d^T \dot{\tau}_d. \end{aligned} \tag{23}$$

Then, (21) is rewritten as

$$\dot{V} = \dot{V}_1 + \dot{V}_2 \le \begin{bmatrix} \varepsilon^T & \tilde{\tau}_d^T \end{bmatrix} \begin{bmatrix} \Gamma_1 & P_1 D \\ D^T P_1 & \Gamma_2 \end{bmatrix} + \gamma^{-2}\|D\|^2 \left\|\tau_d\right\|^2 + \gamma^{-2}\left\|\tau_d\right\|^2 + \gamma^{-2}\left\|\dot{\tau}_d\right\|^2, \tag{24}$$

where

$$\Gamma_1 = \gamma^2 P_1 P_1^T + P_1 A + P_1 Y K_b, \tag{25}$$

$$\Gamma_2 = \gamma^2 \left(P_2 P_2^T + \left(P_2 K_d\right)\left(P_2 K_d\right)^T\right) - P_2 K_d. \tag{26}$$

By using Schur complement formula, it can be concluded that there is a positive scalar satisfying $\begin{bmatrix} \Gamma_1 & P_1 D \\ D^T P_1 & \Gamma_2 \end{bmatrix} < -I$ when $\Gamma < 0$.

So, we have

$$\dot{V} < -\left\|\begin{bmatrix} \varepsilon & \tilde{\tau}_d \end{bmatrix}^T\right\|^2 + \gamma^{-2}\|D\|^2 \left\|\tau_d(t)\right\|^2 + \gamma^{-2}\left\|\tau_d(t)\right\|^2 + \gamma^{-2}\left\|\tau_d(0)\right\|^2. \tag{27}$$

As the iteration increases, the input signal will converge to the specified bound. In addition, the trajectory tracking error will converge to a specific limit. It can be seen that the controller converges, and the actual value always reaches the desired value.

5. Improved B-Spline Optimization

All the classical properties of B-spline basis function can be extended to the multidegree space [20]. In order to calculate spline curves effectively, it is suggested to use multiscale

equations formed by transition functions to calculate each function easily. Therefore, although the resolution of linear systems with as many spatial dimensions is involved, this method has the advantage of locality.

On the basis of cubic uniform B-spline, the motion trajectories of robotic manipulators are classified according to the importance of tasks. Based on the convex hull characteristics of uniform B-spline, the distance information and motion constraint are combined to make the part of trajectory with high-precision converge to smooth, safe, and feasible trajectory in a short time. Suppose that the form of B-spline curve of trajectory can be described as follows:

$$q(t) = \sum_{i=0}^{n-1} p_i B_i^d(t), \quad t_0 \le t \le t_{n-1}, \tag{28}$$

where $B_i^d(t)$ is a B-spline basis function of degree d and the control points p_i must be computed by imposing interpolation conditions on the given data points $q_{i\text{th}}$.

The number of control nodes, the number of times, and the number of node vectors satisfy the formula

$$F = N + d + 1. \tag{29}$$

Then the piecewise polynomial of B-spline basis function is shown in the equation

$$\begin{aligned} &\left[B_{i-d,d+1}(s(t)) \;\; B_{i-d+1,d+1}(s(t)) \;\; \cdots \;\; B_{i,d+1}(s(t)) \right] \\ &= \left[1 \;\; s(t) \;\; s^2(t) \;\; \cdots \;\; s^d(t) \right] F^{d+1}, \end{aligned} \tag{30}$$

where F^{d+1} is the constant matrix determined by the degree.

Then the optimized path $P(s(t))$ is shown in the following equation:

$$P(s(t)) = \left[1 \;\; s(t) \;\; s^2(t) \;\; s^3(t) \right] F^4 \begin{bmatrix} Q_{i-3} \\ Q_{i-2} \\ Q_{i-1} \\ Q_i \end{bmatrix}. \tag{31}$$

According to the actual scene, the joint angular acceleration curve often has the situation that the starting and ending positions are not zero and the third derivative is discontinuous, which will produce vibration and mechanical impact during the movement of the manipulator and affect the stability of the controller signal. In order to solve the above problems, the constraint conditions are added to each segment to modify the signal.

The function expression of cubic B-spline interpolation function can be expressed as formula (31) after adding correction function to each segment interval

$$P_i(t) = P(s(t)) + \kappa_i(t), \tag{32}$$

where $\kappa_i(t)$ is the modified function and $P_i(t)$ is the improved interpolation function. The motion process of the robotic manipulator can be divided into the three following parts:

Part 1: the correction function in the first subsection $\kappa_i(t)$ is to satisfy that the acceleration of the robotic manipulator is stationary at the beginning, and the other properties of cubic spline interpolation remain unchanged. $\kappa_i(t)$ should satisfy the following:

$$\begin{cases} \kappa_1(0) = \kappa_1'(0) = 0, \\ \kappa_1''(0) = -P''(s(0)), \\ \kappa_1(1) = \kappa_1'(1) = \kappa_1''(1) = 0. \end{cases} \tag{33}$$

Through the constraints of formula (33), we can obtain the modified function in the first interval as

$$\kappa_1(t) = \frac{Q_0}{2} t^2 (t-1)^3. \tag{34}$$

Part 2: in the second part, which represents the process of the robotic manipulator in the process of machining, we mainly consider the discontinuity of the third derivative in this part. $\kappa_i(t)$ should satisfy the constraints as

$$\begin{cases} \kappa_n(t_{n-1}) = \kappa_n'(t_{n-1}) = \kappa_n''(t_{n-1}) = 0, \\ \kappa_n(t_n) = \kappa_n'(t_n) = \kappa_n''(t_n) = 0, \\ \kappa_{n-1}'''(t_n - 0) + P_{n-1}'''(t) = \kappa_{n-1}'''(t_n + 0) + P_n'''(t). \end{cases} \tag{35}$$

Through the constraints of formula (35), we can obtain the modified function in the second interval as

$$\kappa_m(t) = \frac{-P_{m-1}'''(t) + P_m'''(t) - 3Q_0}{6} (t - t_{m-1})^3 (t - t_m)^3. \tag{36}$$

Part 3: the correction factor in the last part keeps the third derivative continuous at the left end of the interval in order to make the acceleration of robotic manipulator stationary at the termination time, and other properties of cubic spline interpolation remain unchanged. $\kappa_i(t)$ should satisfy the constraints as

$$\begin{cases} \kappa_i(t_{i-1}) = \kappa_i'(t_{i-1}) = \kappa_i''(t_{i-1}) = 0, \\ \kappa_i(t_i) = \kappa_i'(t_i), \\ \kappa_i''(t_i) = -P''(s_i), \\ \kappa_{i-1}'''(t_{i-1} - 0) + P_{i-1}'''(t) = \kappa_{n-1}'''(t_n + 0) + P_i'''(t). \end{cases} \tag{37}$$

Through the constraints of formula (37), we can obtain the modified function in the third interval as

$$\kappa_i(t) = \frac{-3Q_0 + F}{6F^4} (t - t_{i-1})^3 (t - t_i)^2 \left(x - \frac{3Q_i t_{i-1} + F}{3Q_i + F} \right). \tag{38}$$

6. Simulation

In this section, the manipulator model is established by using MATLAB 2020a, and the designed controller is simulated. In order to show the effect of the proposed method in high-order iteration, the following simulations extract the data after 80 iterations. The parameters of the manipulator are shown in Table 1.

Firstly, the distribution of control points is adjusted according to the importance of the task, which makes the

Table 1: Simulation parameters of the manipulator in Figure 1.

m_1 (kg)	5.0	m_2 (kg)	3.0
l_1 (m)	1.0	I_1 (kg·m^2)	2.0
l_2 (m)	1.0	I_2 (kg·m^2)	5.0

distribution gap between the three parts larger. By showing the trajectory tracking effect before and after the improved B-spline optimization method, the application value of the improved scheme is illustrated.

As shown in Figures 3 and 4 , the improved B-spline optimization method can effectively improve the trajectory tracking effect of the robotic manipulator. On the premise of nonuniform distribution of control points, part 1 and part 3 with fewer control points can still keep tracking accuracy under the action of correction function. The root mean square errors (RMSE) are 3.6% and 0.5%, respectively, from the error data in Figures 3(d) and 4(d).

Next, the B-spline optimization scheme with uniform distribution of control points is simulated.

As shown in Figures 5 and 6 , the improved B-spline optimization method can also effectively improve the trajectory tracking accuracy under the condition that the common control points are evenly distributed. Due to the uniform distribution of control points, the accuracy of part 1 and part 3 is improved, so the overall error decreases. Their RMSE are 2.9% and 0.4%, respectively.

The above is to discuss the signal repetition accuracy and trajectory tracking error of the controller. Finally, we will show the motion effects of the robotic manipulator using the improved B-spline optimization scheme under different conditions. In this section, we set the parameters of three different robotic manipulators. Their differences are different in length and mass, as shown in Table 2. The parameters not involved are the same as those in Table 1.

Case 1 uses the same parameters as those in Table 1. As shown in Figure 7, through the simulation results of the robotic manipulator's motion trajectory, it can be seen that the improved B-spline optimization scheme can make the manipulator's motion trajectory smoother and has better suppression effect on interference and vibration. In Case 2, the length of mechanical arm is changed, as shown in Figure 8. It can still be seen that the trajectory smoothness of the manipulator is significantly improved after using the scheme proposed in this paper. However, as shown in Figure 9, the improved B-spline optimization scheme does not achieve obvious results in example 3. Compared with the actual manipulator control system and the experiment of the Simscape model, because the stiffness of the material is not considered in the simulation, the change of the weight only affects the inertia and reduces the vibration and interference. Therefore, the optimization effect of Case 3 simulation is not obvious.

In order to show the ability of the proposed algorithm in practical application as much as possible, a 2-DOF robot model is established by using MATLAB/Simscape to verify the performance of the proposed algorithm in repetitive control and compare with the conventional B-spline and rational B-spline optimization algorithms [29]. Due to the limitation of hardware, more parameters are introduced in Simscape, which leads to the slow operation speed. Therefore, the trajectory of the manipulator after 10 iterations is extracted.

Firstly, the coordinate data of the given trajectory is extracted, and the constraint conditions are fitted according to the dynamic model of the manipulator and the physical model built in Simscape. Next, B-spline interpolation method is used to smooth the trajectory coordinates and convert them into servo motor control signals, which are provided to the joint controller. Finally, visualization processing is carried out to show the motion in MATLAB, and the proposed interpolation compensation method is used to correct the error before entering the next iteration. The process is shown in Figure 10.

The parameters of joint rotation friction can be added to each joint of the quasi-physical model to make the virtual manipulator closer to the real manipulator. The quasi-physical model of manipulator based on Simscape is constructed, including all the revolute joints with rotating friction and their parameters. The model of revolute joint in Simscape is shown in Figure 11.

The parameters of the manipulator model established in Matlab/Simscape are from the data of the spraying manipulator provided by the Vehicle Research Institute, as shown in Table 3.

Figure 12 shows the coordinates of the trajectory of the conventional B-spline interpolation method, rational B-spline optimization algorithm, and the algorithm proposed in this paper under the Simscape model. In order to highlight the performance of the algorithm more clearly, it continued to run for a period of time, and the repeated loop interval with obvious errors was selected for analysis.

As shown in Figure 12, the input command is obtained by directly connecting the calibration points, and the track line is not smoothed. From the simulation data, it can be seen that the trajectory of the manipulator control output is gradually smoothed by the spline interpolation optimization method proposed in this paper. Through the Simscape simulation results, it is shown that, in the actual situation, due to the existence of interference, after 20 iterations, the tracking accuracy is reduced, but the method proposed in this paper can still ensure excellent tracking accuracy on the premise of keeping the track smooth. Although the relative error of rational B-spline optimization method is small, there is obvious trajectory breakpoint problem due to the small number of control points and not considering the control points at the turning. Due to the uniform distribution of control points and simple trajectory, the traditional B-spline

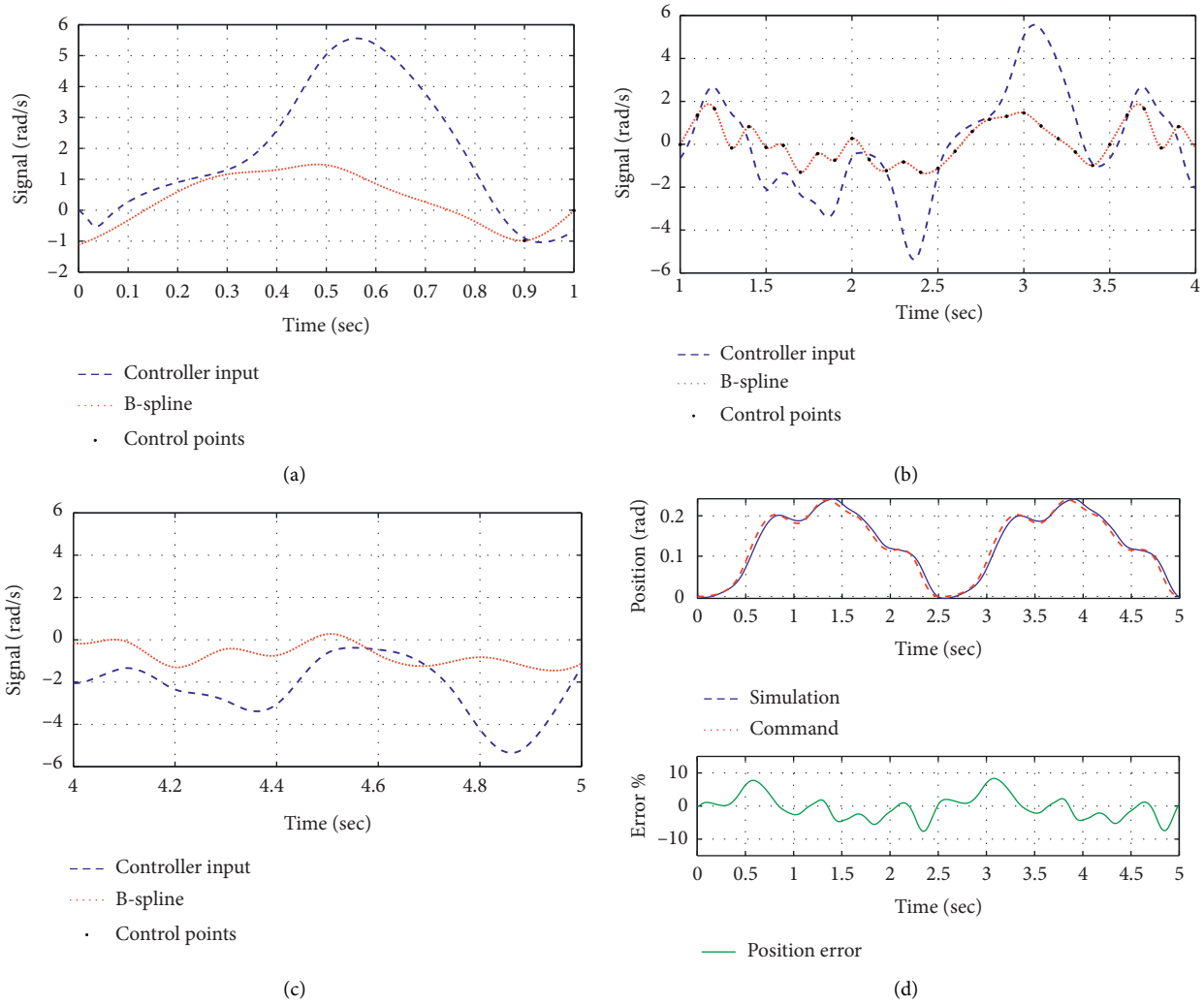


Figure 3: Input modification scheme of manipulator controller based on conventional B-spline optimization (control points with nonuniform distribution). (a) Controller input and compensator output in part 1. (b) Controller input and compensator output in part 2. (c) Controller input and compensator output in part 3. (d) Comparison and error of reference position and measurement position.

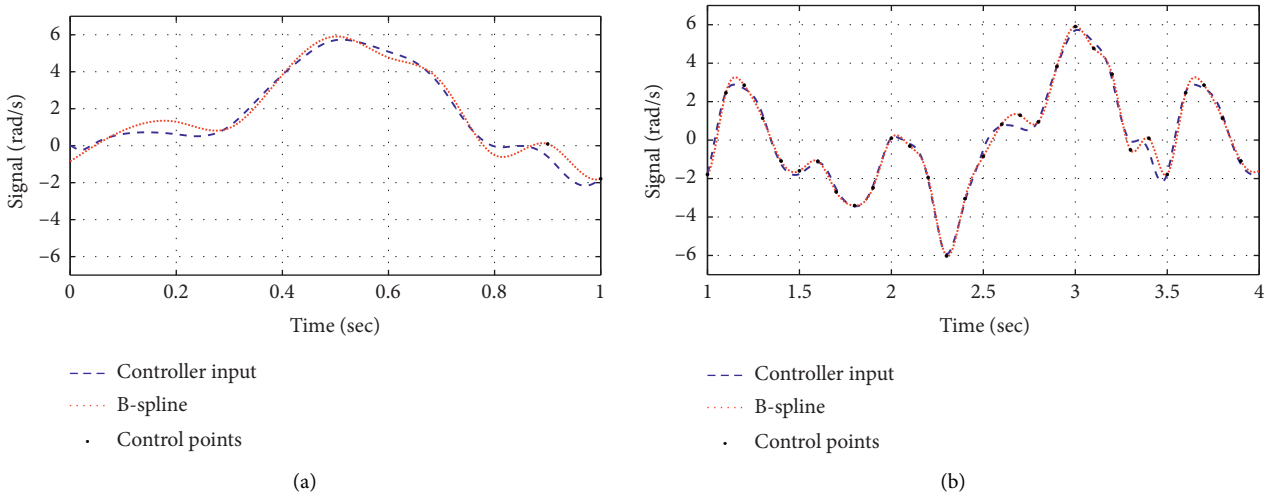


Figure 4: Continued.

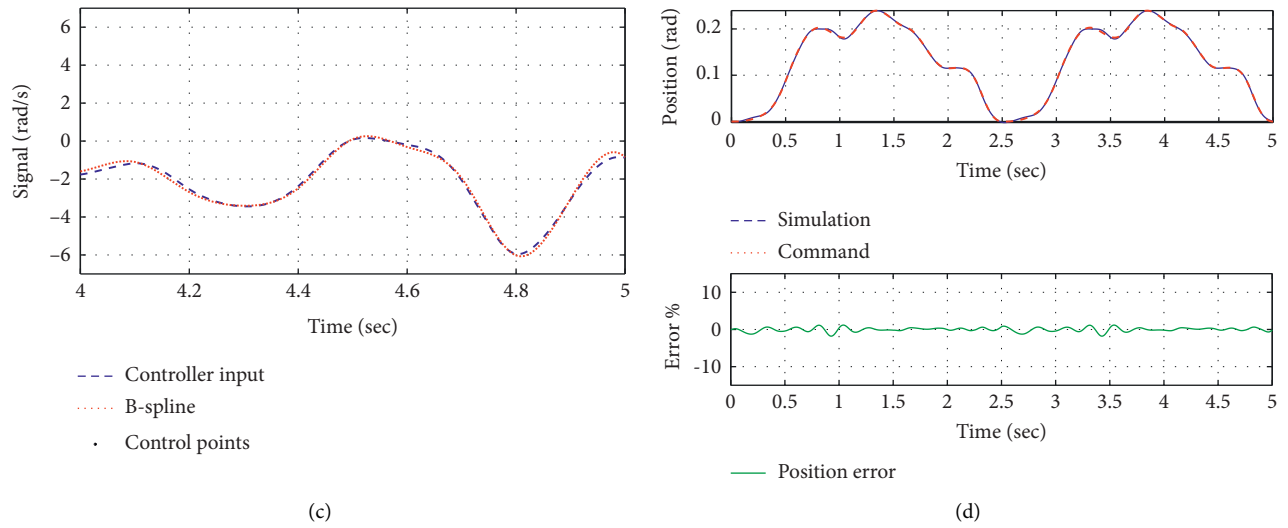


Figure 4: Input modification scheme of manipulator controller based on improved B-spline optimization (control points with nonuniform distribution). (a) Controller input and compensator output in part 1. (b) Controller input and compensator output in part 2. (c) Controller input and compensator output in part 3. (d) Comparison and error of reference position and measurement position.

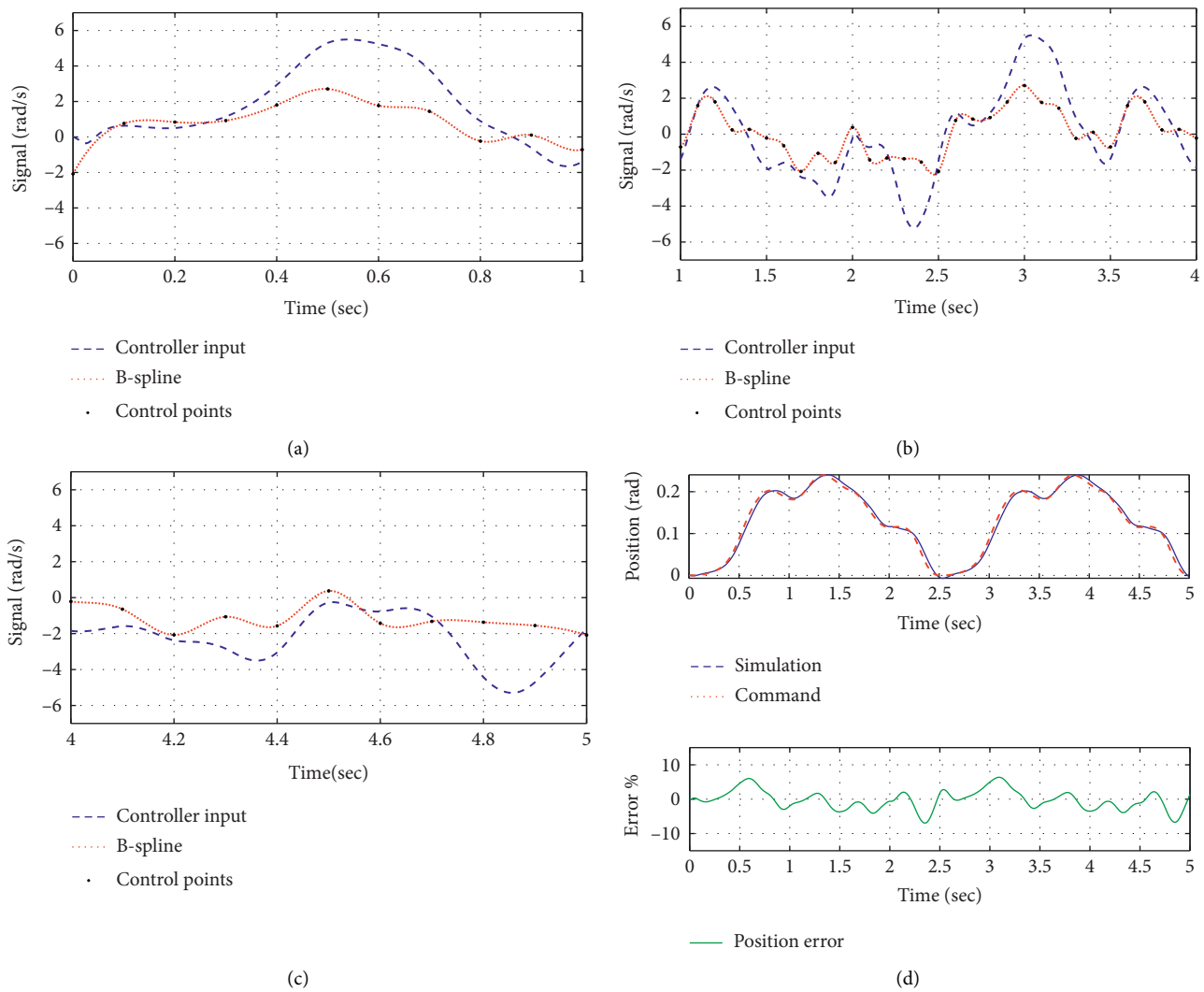


Figure 5: Input modification scheme of manipulator controller based on conventional B-spline optimization. (a) Controller input and compensator output in part 1. (b) Controller input and compensator output in part 2. (c) Controller input and compensator output in part 3. (d) Comparison and error of reference position and measurement position.

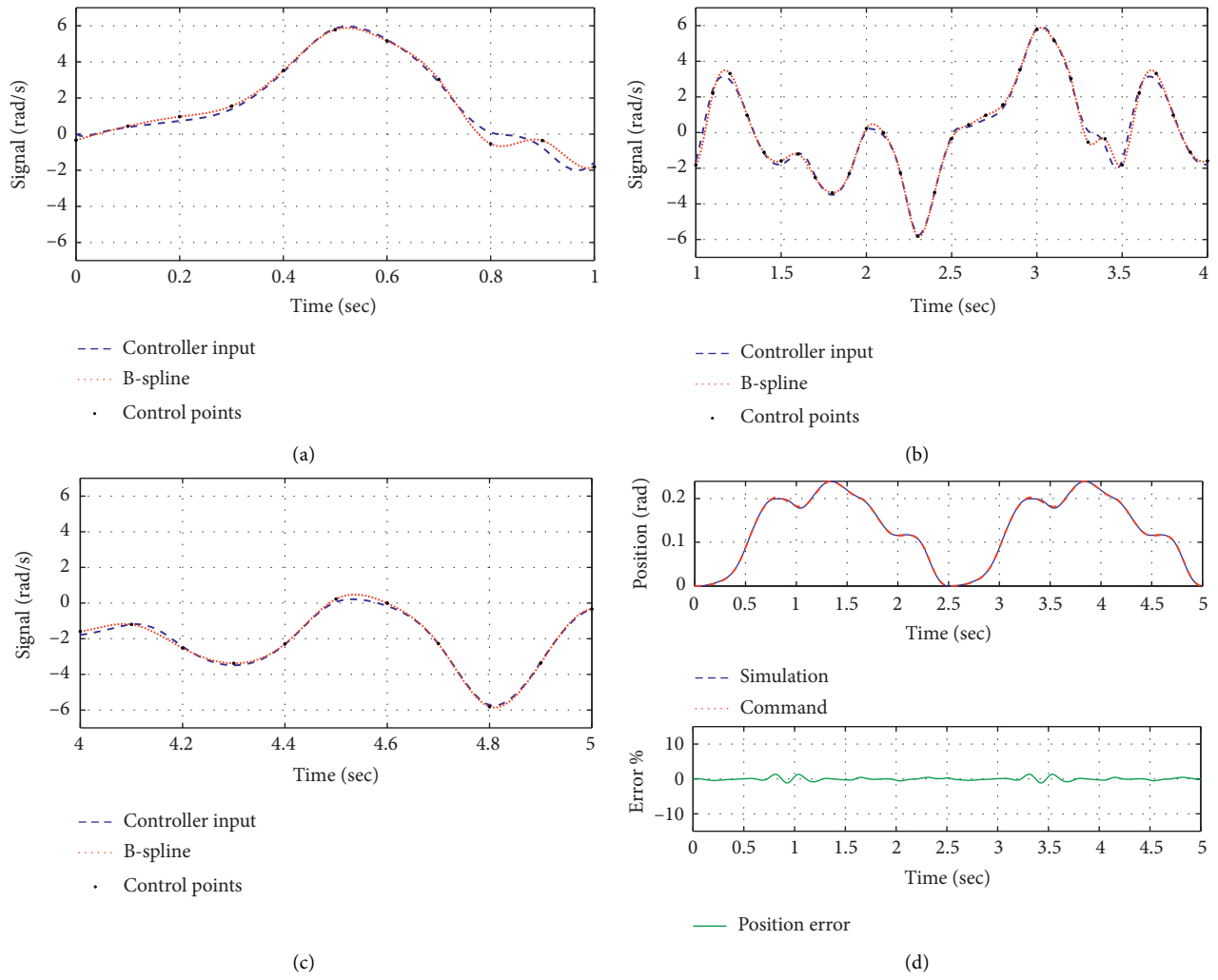


Figure 6: Input modification scheme of manipulator controller based on improved B-spline optimization. (a) Controller input and compensator output in part 1. (b) Controller input and compensator output in part 2. (c) Controller input and compensator output in part 3. (d) Comparison and error of reference position and measurement position.

Table 2: Simulation parameters of the manipulator in different cases.

	Case 1	Case 2	Case 3
m_1 (kg)	5	5	5
m_2 (kg)	3	3	
l_1 (m)	1	1	1
l_2 (m)	1	0	5

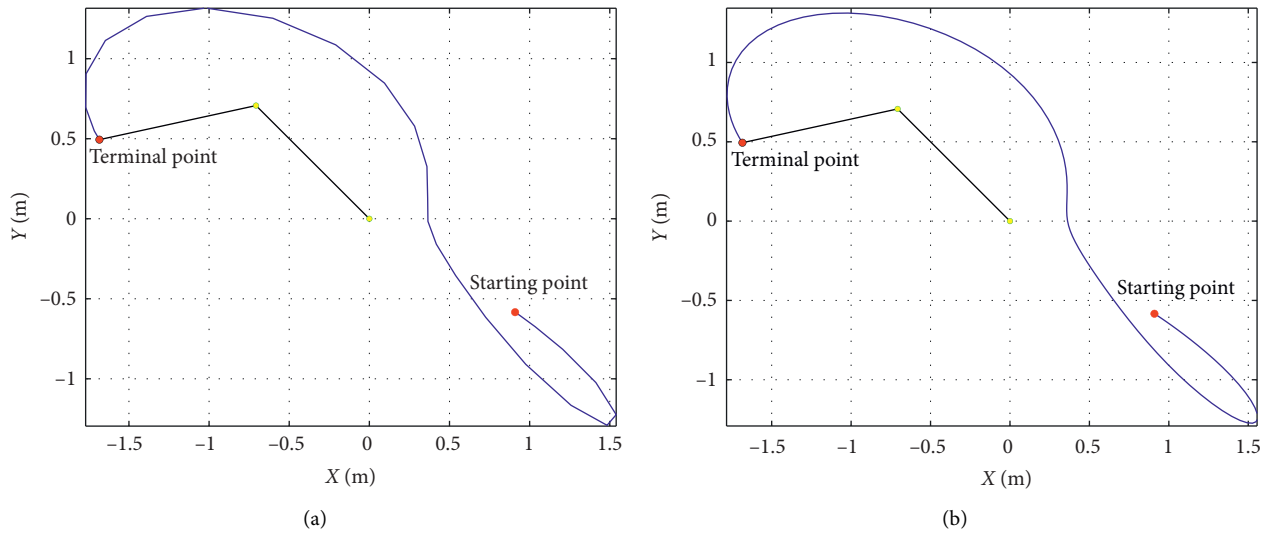


Figure 7: The trajectory of 2-DOF robotic manipulator in case 1. (a) Conventional B-spline optimization. (b) Improved B-spline optimization.

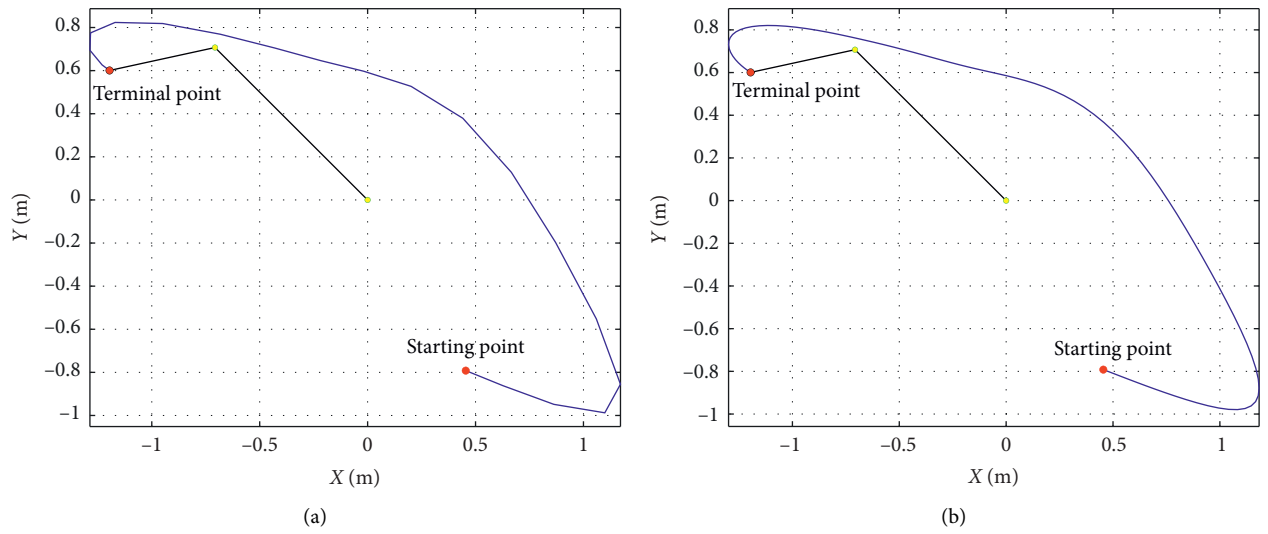


Figure 8: The trajectory of 2-DOF robotic manipulator in case 2. (a) Conventional B-spline optimization. (b) Improved B-spline optimization.

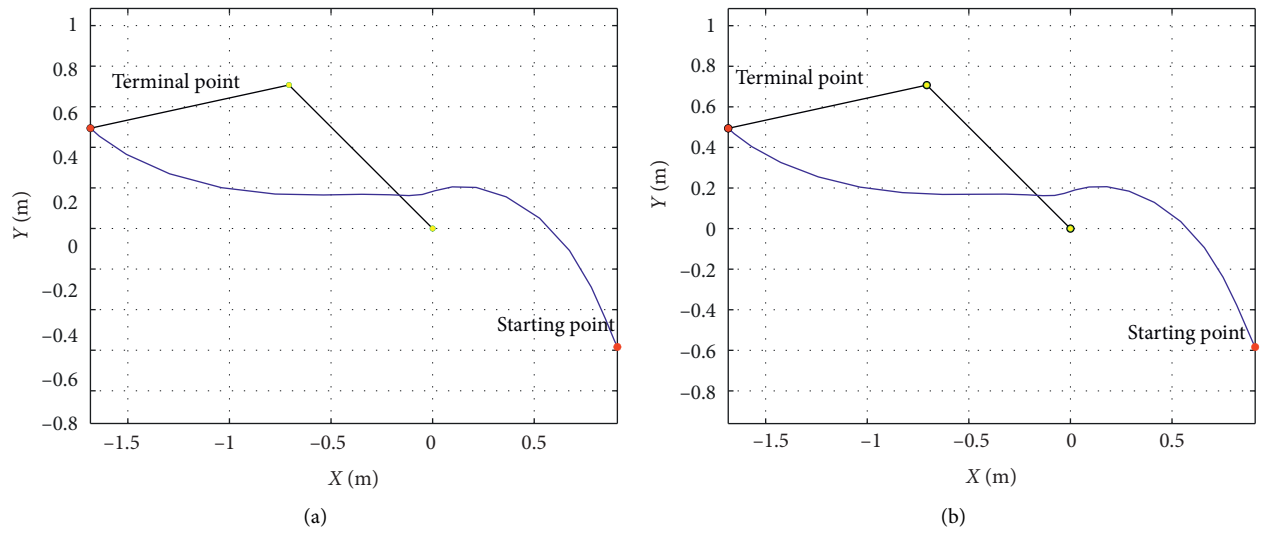


Figure 9: The trajectory of 2-DOF robotic manipulator in case 3. (a) Conventional B-spline optimization. (b) Improved B-spline optimization.

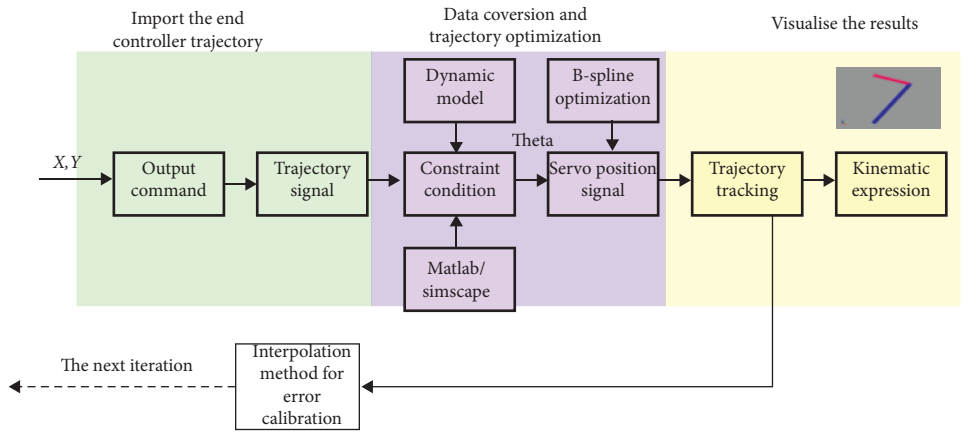


Figure 10: Repetitive control simulation process.

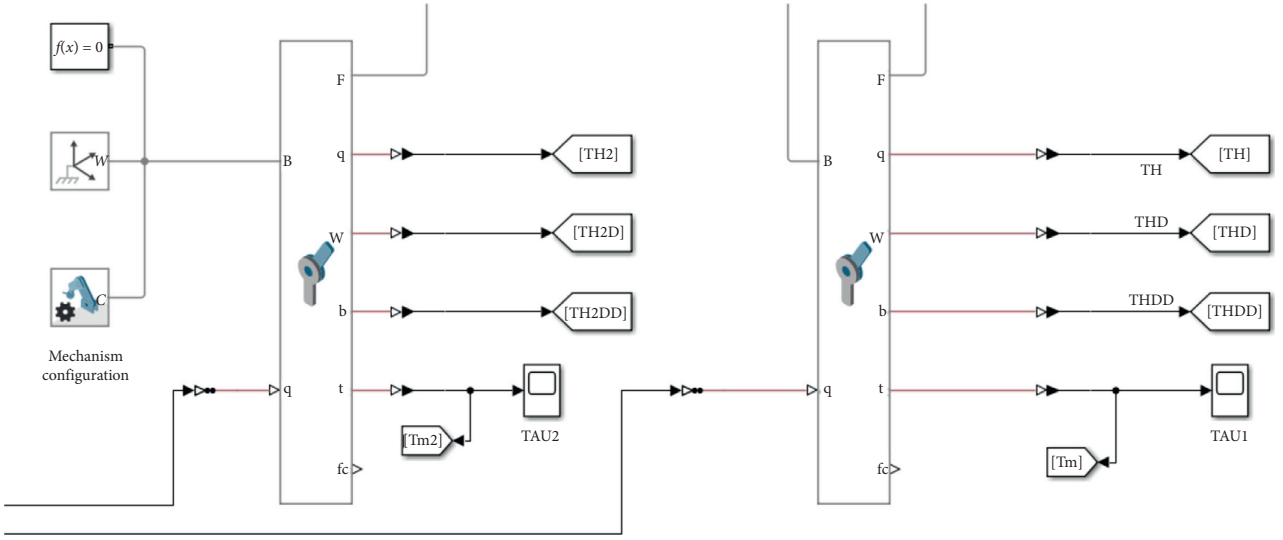


Figure 11: Revolute joint model in Simscape.

Table 3: Parameters of revolute joint in Simscape.

Joint	Values of mass (kg)	Breakaway friction torque (Nm)	Breakaway friction velocity (rad/s)	Coulomb friction torque (Nm)	Viscous friction coefficient (Nm/(rad/s))
1	4.077	30×10^{-3}	0.01	24×10^{-3}	0.001
2	1.832	10×10^{-4}	0.01	5×10^{-4}	0.001

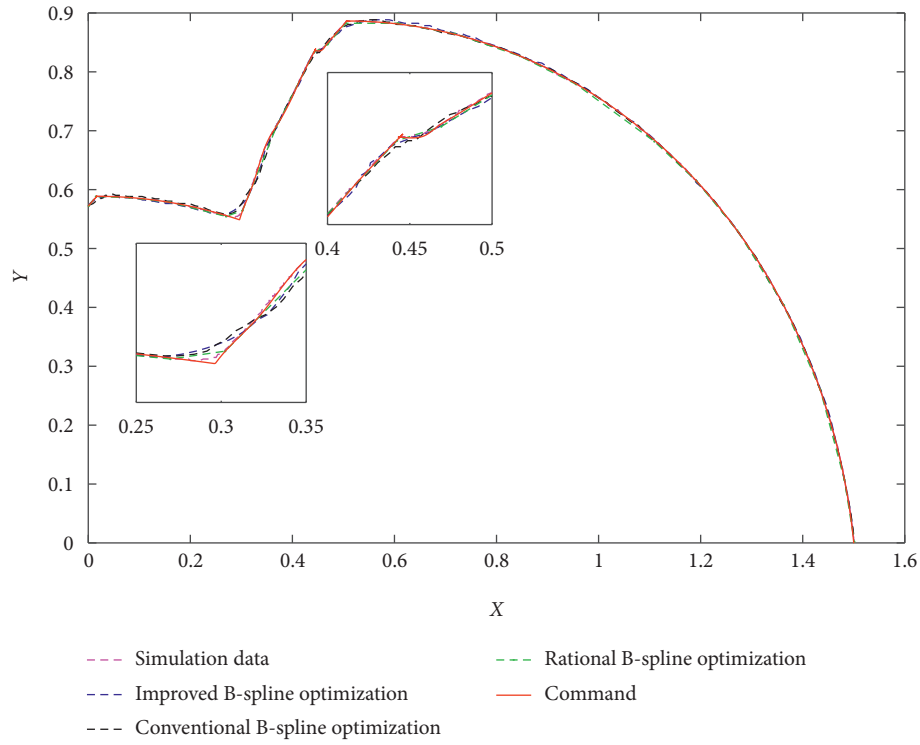


Figure 12: Comparison of simulation results in Simscape.

interpolation method is partially optimal, but there is obvious vibration.

7. Conclusions

The application of iterative control in the control system of robotic manipulator can facilitate decision-making and adjust the gap between the repeated motion results and the expected results, so as to meet the requirements of repeated motion. However, the control process of robotic manipulator is highly nonlinear, strongly coupled, and time-varying. Therefore, the combination of iterative learning control and other intelligent control methods is used to optimize the control. It can improve the robustness and adaptability of the system and achieve good tracking error convergence and high tracking accuracy. In this paper, three methods are proposed to compensate the motion characteristics of the target in different intervals according to the motion characteristics of the target. This scheme can realize the control of the nonlinear strong coupling dynamic uncertain system in a given time range through a very simple algorithm and obtain the high-precision tracking of the given trajectory and realize the stability and robustness of the system. By combining MATLAB simulation into the iterative learning control system, the effectiveness of the improved B-spline repetitive control method can be quickly verified. Thus, the application efficiency of the new method in the actual control system can be improved. The simulation results show that the improved cubic B-spline optimization method has better tracking accuracy than the conventional cubic B-spline optimization method. However, in some cases, due to the higher order of interpolation method, the Runge phenomenon will appear near the end point, resulting in the failure to return to the end point coordinates correctly. After many iterations, this problem will gradually become serious. Some references have proposed solutions for Runge phenomenon [30, 31], and we will implement these solutions in the future robotic manipulator control system.

Notations

q: Angular position
$\mathbf{f}$: Friction matrix
$\dot{q}$: Angular velocity
τ_d: Uncertain external disturbances
$\ddot{q}$: Angular acceleration
l: Length of the connecting rod
$\mathbf{M}$: Positive definite inertia matrix
m: Mass of the connecting rod
$\mathbf{C}$: Coriolis force matrix
K: Gain of observer
$\mathbf{g}$: Gravity matrix
I: Moment of inertia
τ: Torque B
B: B-spline basis function.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Liaoning Province Science and Technology Major Project (2020020304-JH1/101) and National Key R&D Program of China (2017YFB1300900).

References

[1] S. Arimoto, M. Sekimoto, and S. Kawamura, "Iterative learning of specified motions in task-space for redundant multi-joint hand-arm robots," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2867–2873, IEEE, Seoul, South Korea, June 2001.

[2] S. Arimoto, M. Sekimoto, and K. Tahara, "Iterative learning without reinforcement or reward for multi joint movements: a revisit of bernstein's DOF problem on dexterity," *Journal of Robotics*, vol. 2010, Article ID 217867, 15 pages, 2010.

[3] T. Inoue, M. Nakano, T. Kubo, S. Matsumoto, and H. Baba, "High accuracy control of a proton synchrotron magnet power supply," *IFAC Proceedings Volumes*, vol. 14, no. 2, pp. 3137–3142, 1981.

[4] T. Inoue, M. Nakano, and S. Iwai, "High accuracy control of servo mechanism for repeated contouring," in *Proceedings of the 10th Annual Symposium on Incremental Motion Control Systems and Devices*, pp. 258–292, Rosemont IL, USA, June 1981.

[5] L. Cuiyan, Z. Dongchun, and Z. Xianyi, "A survey of repetitive control," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 1160–1166, Sendai, Japan, July 2004.

[6] 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.

[7] D. H. Wang and S. Zhang, "Improved neural network-based adaptive tracking control for manipulators with uncertain dynamics," *International Journal of Advanced Robotic Systems*, vol. 17, no. 4, Article ID 1729881420947562, 2020.

[8] A. Azizi, "A case study on designing a sliding mode controller to stabilize the stochastic effect of noise on mechanical structures: residential buildings equipped with ATMD," *Complexity*, vol. 2020, Article ID 9321928, 17 pages, 2020.

[9] Z. Mu, T. Liu, W. Xu, Y. Lou, and B. Liang, "Dynamic feedforward control of spatial cable-driven hyper-redundant manipulators for on-orbit servicing," *Robotica*, vol. 37, no. 1, pp. 18–38, 2019.

[10] Y. Dai, S. Yu, Y. Yan, and X. Yu, "An EKF-based fast tube MPC scheme for moving target tracking of a redundant underwater vehicle-manipulator system," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 6, pp. 2803–2814, 2019.

[11] W. Xiang and S. Yan, "Dynamic analysis of space robot manipulator considering clearance joint and parameter uncertainty: modeling, analysis and quantification," *Acta Astronautica*, vol. 169, pp. 158–169, 2020.

[12] S. R. Friedrich and M. Buss, "Parameterizing robust manipulator controllers under approximate inverse dynamics: a double-Youla approach," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 15, pp. 5137–5163, 2019.

[13] X. Lu and Y. Jia, "Adaptive coordinated control of uncertain free-floating space manipulators with prescribed control performance," *Nonlinear Dynamics*, vol. 97, no. 2, pp. 1541–1566, 2019.

[14] S. Wang, X. Ren, J. Na, and T. Zeng, "Extended-state-observer-based funnel control for nonlinear servomechanisms with prescribed tracking performance," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 98–108, 2016.

[15] J. Na, A. S. Chen, G. Herrmann, R. Burke, and C. Brace, "Vehicle engine torque estimation via unknown input observer and adaptive parameter estimation," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 409–422, 2017.

[16] Z. Li, C. Yang, C. Y. Su, S. Deng, F. Sun, and W. Zhang, "Decentralized fuzzy control of multiple cooperating robotic manipulators with impedance interaction," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 4, pp. 1044–1056, 2014.

[17] Y. Fan, K. Xing, and X. Jiang, "Fuzzy adaptation algorithms' control for robot manipulators with uncertainty modelling errors," *Complexity*, vol. 2018, Article ID 5468090, 8 pages, 2018.

[18] Y. Wang, K. Zhu, B. Chen, and M. Jin, "Model-free continuous nonsingular fast terminal sliding mode control for cable-driven manipulators," *ISA Transactions*, vol. 98, pp. 483–495, 2020.

[19] Y. Wang, S. Li, D. Wang, F. Ju, B. Chen, and H. Wu, "Adaptive time-delay control for cable-driven manipulators with enhanced nonsingular fast terminal sliding mode," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 3, pp. 2356–2367, 2020.

[20] B. Xiao, S. Yin, and O. Kaynak, "Tracking control of robotic manipulators with uncertain kinematics and dynamics," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 10, pp. 6439–6449, 2016.

[21] M. Latifinavid, A. Donder, and E. i. Konukseven, "High-performance parallel hexapod-robotic light abrasive grinding using real-time tool deflection compensation and constant resultant force control," *The International Journal of Advanced Manufacturing Technology*, vol. 96, no. 9–12, pp. 3403–3416, 2018.

[22] J. Meng, S. Wang, G. Li et al., "Iterative-learning error compensation for autonomous parking of mobile manipulator in harsh industrial environment," *Robotics and Computer-Integrated Manufacturing*, vol. 68, Article ID 102077, 2021.

[23] L. Biagiotti, L. Moriello, and C. Melchiorri, "Improving the accuracy of industrial robots via iterative reference trajectory modification," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 3, pp. 831–843, 2020.

[24] A. Gasparetto and V. Zanotto, "A new method for smooth trajectory planning of robot manipulators," *Mechanism and Machine Theory*, vol. 42, no. 4, pp. 455–471, 2007.

[25] Y. Li, T. Huang, and D. G. Chetwynd, "An approach for smooth trajectory planning of high-speed pick-and-place parallel robots using quintic B-splines," *Mechanism and Machine Theory*, vol. 126, pp. 479–490, 2018.

[26] K. Basinski, B. Ufnalski, and L. M. Grzesiak, "Particle swarm based repetitive spline compensator for servo drives," *Przegl. Elektrotechn*, vol. 1, no. 2, pp. 181–187, 2017.

[27] S. Qian, K. Bao, B. Zi, and W. D. Zhu, "Dynamic trajectory planning for a 3-DOF cable-driven parallel robot using quintic B-splines," *Journal of Mechanical Design*, vol. 142, no. 7, 2019.

[28] K.-H. Yu, Y. Shito, and H. Inooka, "Position control of an underactuated manipulator using joint friction," *International Journal of Non-linear Mechanics*, vol. 33, no. 4, pp. 607–614, 1998.

[29] Y. L. Kuo, C. C. Lin, and Z. T. Lin, "Dual-optimization trajectory planning based on parametric curves for a robot manipulator," *International Journal of Advanced Robotic Systems*, vol. 17, no. 3, Article ID 1729881420920046, 2020.

[30] C. Ye, S. Feng, Z. Xue, C. Guo, and Y. Zhang, "Defeating runge problem by coefficients and order determination method with various approximation polynomials," in *Proceedings of the 37th Chinese Control Conference (CCC)*, pp. 8622–8627, Wuhan, China, July 2018.

[31] D. Chen, "Solving the problem of runge phenomenon by pseudoinverse cubic spline," in *Proceedings of the 2014 IEEE 17th International Conference on Computational Science and Engineering*, IEEE, Chengdu, China, December 2014.

Hindawi
Complexity
Volume 2021, Article ID 6628802, 14 pages
https://doi.org/10.1155/2021/6628802

Research Article

Development and Evaluation of a Rehabilitation Wheelchair with Multiposture Transformation and Smart Control

Wujing Cao[1,2] **Hongliu Yu**[3] **Xinyu Wu**[1,2] **Sujiao Li**[3] **Qiaoling Meng**[3] **and Chunjie Chen**[1,2]

[1]*Guangdong Provincial Key Lab of Robotics and Intelligent System, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518005, China*
[2]*CAS Key Laboratory of Human-Machine Intelligence-Synergy Systems, Shenzhen Institutes of Advanced Technology, Shenzhen 518005, China*
[3]*Rehabilitation Engineering and Technology Institute, University of Shanghai for Science and Technology, Shanghai 200093, China*

Correspondence should be addressed to Hongliu Yu; yhl98@hotmail.com and Xinyu Wu; xy.wu@siat.ac.cn

Received 10 October 2020; Accepted 22 February 2021; Published 27 February 2021

Academic Editor: A Zabihollah

Stroke and other neurological disorders have an effect on mobility which has a significant impact on independence and quality of life. The core rehabilitation requirements for patients with lower limb motor dysfunction are gait training, restand, and mobility. In this work, we introduce a newly developed multifunctional wheelchair that we call "ReChair" and evaluated its performance preliminarily. ReChair seamlessly integrates the mobility, gait training, and multiposture transformation. ReChair driving and multiposture transformation are done using the voice, button, and mobile terminal control. This work describes the functional requirements, mechanical structure, and control system and the overall evaluation of ReChair including the kinematic simulation of the multiposture transformation and passive lower limb rehabilitation training to quantitatively verify the motion capability of ReChair, the voice control system evaluation that shows how the voice recognition system is suitable for home environment, the sensorless speed detection test results that indicate how the wheel speeds measured by sensorless method are appropriate for travelling control, and the passive and balance training test results that show how the lower limb rehabilitation training in daily life by ReChair is convenient. Finally, the experimental results show that ReChair meets the patients' requirements and has practical significance. It is cost-effective, easy to use, and supports multiple control modes to adapt to different rehabilitation phases.

1. Introduction

The aging population trend is accelerating in the world. According to "The World Population Prospects 2019: Highlights," released by the United Nations, one in six people in the world will be over the age of 65 (16%) by 2050, compared with one in 11 in 2019 (9%). The share of the population aged 65 years or more is projected to double between 2019 and 2050 in most countries [1]. Stroke, spinal cord injury (SCI), and other neurological disorders are often related to aging [2] and have an effect on various functions necessary for daily life, especially mobility which has a significant impact on independence and quality of life. The relationship points out that a larger number of seniors are expected to be affected by lower limb motor dysfunction. Additionally, stroke or SCI patients who stay in bed for a long time are at risk for secondary problems including pressure ulcers, obesity, diabetes mellitus, osteoporosis, and other chronic health conditions that increase the risk of mortality [3].

The good news is that lower limb motor function can be improved through rehabilitation training. An effective training method is to conduct high-intensity repetitive tasks [4]. Conventional lower limb rehabilitation training relies on manual exercises, in which the physiotherapist and the patient must be in a one-to-one interaction. However, as the

amount of lower limb motor dysfunction patients is increasing, the number of physiotherapists is far from being sufficient [5]. This shortage leads to limited therapy sessions for the patients and physical exertion of the physiotherapist. In order to alleviate the physical burden on physiotherapists and maximize independent function of stroke or SCI patients, numerous lower limb rehabilitation and assistive devices have been proposed. Lower limb rehabilitation devices are typically divided into body weight support training devices (BWSTDs) and lower limb exoskeletons (LLE) [6]. The basic components of a BWSTD are a body weight support system and the LLE linked to a treadmill or movable footplates. The significant size and cost of BWSTDs usually result in limited applications. Additionally, they are all installed in specialized facilities such as hospitals and clinical centers.

To provide daily assistive systems and substitute the residual neurological deficits that prevent the return to upright bipedal ambulation, LLEs are applied in the course of daily rehabilitation [7–9]. LLEs are usually able to carry the person wearing them, providing walking assistive force and keeping the body balance. There are several wearable LLEs currently in development and early clinical evaluation. The LLE ReWalk developed by Argo Medical Technologies provides powered hip and knee motion to enable individuals with SCI to stand upright, walk, turn, and climb stairs up and down. The ankles are passive and use a double-action orthosis with limited motion and spring-assisted dorsiflexion. The walking control is realized through subtle trunk motion and changes in the center of gravity. It is the first exoskeleton to receive FDA clearance for personal and rehabilitation use in the United States [10]. Researchers from Vanderbilt University proposed a powered LLE that provides assistive torques at both hip and knee joints for SCI individuals. A built-in carbon fiber ankle-foot orthosis (AFO) was used to provide ankle stability and transmit the weight of the LLE to the ground. The developed LLE was called Indego in the following research studies [11, 12]. The system developed by Ekso Bionics consists of electric motors to power the movement of hip and knee joints, passive spring-loaded ankle joints, footplates on which the user stands, and a backpack that houses a computer, battery supply, and wired controller [3]. It is necessary for Rewalk, Indego, and Ekso to use supplementary walking aids, such as crutches, to balance and change direction. REX (Rex Bionics PLC, London, UK) is the only commercialized self-stabilizing LLE that requires no supplemental upper body support to balance [13].

The large size and significant cost of existing BWSTDs limit their daily life applications. On the other hand, the level of safety, endurance, and wear comfort leads to limited use of LLEs [14]. The core rehabilitation requirements of patients with lower limb motor dysfunction are gait training, multiposture transformation, and mobility. BWSTDs and LLEs are not irreplaceable means to meet the above requirements. The wheelchairs are still the most used devices for patients with lower limb motor dysfunction. Various research studies about different aspects of powered wheelchairs have been conducted [15–21]. With the development of human-computer interface hardware and sensor processing algorithms, the smart wheelchairs give people with lower limb motor dysfunction not only mobility but also the necessary rehabilitation and daily living help [22–24]. Doyoung and colleagues proposed a wheelchair integrated lower limb rehabilitation system [25] that is composed of an electrical wheelchair, a lifter, and an LLE. The gait training and transportation of the patients are combined by an LLE that is based on a wheelchair. This provides a new train of thought for rehabilitation of patients with lower limb motor dysfunction. However, the gait training of this system relies on the LLE, and this is complex and difficult to use in daily life. Therefore, a mobile device which has the multiposture transformation function to avoid pressure ulcers and that combines the gait training system in daily life would be beneficial to the patients with lower limb motor dysfunction.

We have presented a technical note and gave a general introduction of the voice-controlled wheelchair in previous work [26]. In this work, we present the development of a multifunctional wheelchair (ReChair) that seamlessly integrates the mobility, gait training, and multiposture transformation in daily life conditions detailly. Compared with BWSTDs of lower limb rehabilitation in hospitals, ReChair provides the benefits of reduced efforts of the nursing staff, being time saving and the ability to be used at home. Compared with LLEs, ReChair is easier and safer in daily use. Firstly, we introduce the main design concepts and technical characteristics of ReChair. Secondly, the detailed specifications of each module including mechanical structure, kinematic simulation, and control system are described. Thirdly, we present the experimental tasks and evaluation of ReChair from different perspectives. The system is discussed and compared with previous related work. Lastly, a brief conclusion and the plan for future studies are given in the final part.

2. Materials and Methods

2.1. Functional Requirements of ReChair. ReChair is the combination of the lower limb rehabilitation training device and powered wheelchair. Four main functional requirements should be satisfied. First of all, ReChair has the basic functions of a powered wheelchair. When the lower limb rehabilitation training is not conducted, ReChair is used as an ordinary powered wheelchair both indoors and outdoors. The electric drive and multiposture transformation function are necessary. These functions are assistive to lower limb rehabilitation training, such as balance control training during standing posture and passive training during sitting or lying position. Then, multiple lower limb training modes are needed for different rehabilitation procedures. The training modes mainly include active and passive training. Active training is realized using the following method. During standing posture, the patient moves the center of gravity left and right. Pressure sensors on the footplate collect information about the center of gravity to control the bird's left and right movements in the game. When the bird eats the fallen gem, the patient scores. Passive training is carried out during sitting or standing posture to drive lower

limb to simulate walking motion or planned trajectory. Finally, multiple control methods are combined in ReChair. The voice, button, and mobile terminal (mobile phone or smart tablet) control are used for driving and multiposture transformation of ReChair. Multiple control methods meet different requirements of the patients. Patients and nursing staff can choose the appropriate way to operate the ReChair in a convenient way.

2.2. Mechanical Design. ReChair is designed based on the idea of modularization. The system design is shown in Figure 1. The basic components of ReChair are frame module, linear drive module, lower limb rehabilitation training module, cushion module, armrest module, and back module.

In order to ensure stability, the frame uses the rear-wheel drive method. Two drive motors (Motion Technology Electric & Machinery Co., Ltd. Taiwan) whose rated voltage is 24 V and power is 320 W are attached to the left and right of the frame for the wheelchair movement. Two 12-Volt batteries for the whole power system are installed in the back of the frame. This arrangement is useful for the backward movement of the center of gravity, which prevents rollover and rear-tipping of the wheelchair. The lower limb rehabilitation training module is connected to the seat module by the connection between the shank and the seat. The linear actuator is used to drive the back and forth motion of the shank. The lower limb passive training is realized by the cooperation of the back/forth and up/down motors of the shank. Force sensors are installed in the footplate for balance training. The front antitilt wheel under the footplate is used as driving safety assistance. The linear drive module mainly includes eight motors for multiposture transformation and lower limb rehabilitation training. The wheelchair can be used for standing, sitting, and lying positions' transformation. Passive training activated by the shank or thigh is also realized by corresponding motors. When the thigh up/down motor moves, it can drive the thigh joint to rotate around its fixed rotation center, so as to lift the patient's thigh and drive the patient to perform lower limb rehabilitation training. The thigh bondage is used to secure the patient's thigh and prevent the patient from slipping while in standing posture. The seat cushion and the wheelchair frame are connected by fixed hinges. When the wheelchair is in the sitting position, the slot on the cushion can be stuck downwards on the side bar of the wheelchair frame so that the frame can support the cushion. The height and the front and rear position of the arm pad are adjustable. The front/back and left/right distance of the controller from the arm pad can also be adjusted to suit different human bodies. The left and right handrails can be rolled backward and removed if needed. These adjustable designs help the patients to get in and out of the wheelchair easily.

Since the wheelchair is mainly used in patients with lower limb dysfunction, the safety and stability of the mechanism are very important. The motion mechanism of the ReChair is briefly shown as Figure 2. The stand-up posture was realized by the linear displacement of the motor and the clockwise rotation of the cushion and back modular. The thigh up and down motions were driven by linear motor and the rotation of the cushion. The linear motor retracts and back modular rotates to realize the transformation from sit to back lie. The linear motor drives the lower limb rehabilitation training modular to realize the shank up and down. The shank back and forth was achieved by the linear displacement of the linear motor parallel to the shank. In order to prevent accidents, the uniqueness of the mechanism motion must be verified. The conditions that the mechanism has definite motion are mechanism DOF is greater than zero and the number of driving link is equal to the mechanism DOF [27]. The planar motion mechanism has three DOFs when the component is not connected with other mechanism. The lower pair introduces two constraints, and the higher pair brings one.

The calculating formula of mechanism DOF is described as follows:

$$F = 3n - 2p_l - p_h, \tag{1}$$

where F is the mechanism DOF, n is the number of moving link, p_l is the number of lower pair, and p_h is the number of higher pair. We calculate the mechanism DOF of each motion mechanism from Figure 3. All the mechanism DOF is equal to one and the same with the number of driving link. The uniqueness of the motion mechanism of the ReChair is verified.

$$\begin{aligned} F_{3(a)} &= 3 \times 3 - 2 \times 4 - 0 = 1, \\ F_{3(b)} &= 3 \times 3 - 2 \times 4 - 0 = 1, \\ F_{3(c)} &= 3 \times 3 - 2 \times 4 - 0 = 1, \\ F_{3(d)} &= 3 \times 3 - 2 \times 4 - 0 = 1, \\ F_{3(e)} &= 3 \times 1 - 2 \times 1 - 0 = 1. \end{aligned} \tag{2}$$

2.3. Kinematic Simulation. The motion modes of ReChair include sit to stand, sit to lie, and lower limb rehabilitation training during sitting. In order to demonstrate that ReChair can realize multiposture transformation and passive lower extremity training in a stable way, the COSMOS motion module of SolidWorks is used to carry out kinematic simulation of the rehabilitation wheelchair. The position of monitoring points and sit to stand procedure is shown in Figure 3(a). When the wheelchair is changing from sitting to standing posture, the driving links are stand-up motor, lie motor, and shank back/forth motor. Before standing, let the front antitilt wheel touch the ground to ensure the stability of the mechanism and prevent its center of gravity from moving forward and causing the unexpected situation or discomfort given the ground impact. The horizontal angular displacement of seat, back, and leg support is shown in Figure 3(b). The whole simulation process is smooth, and there is no interference in the motion of the mechanism. When the displacement of the standing motor reaches the limit position, the wheelchair is exactly at the desired posture. The design requirements are achieved, and the motor selection and structural design are reasonable.

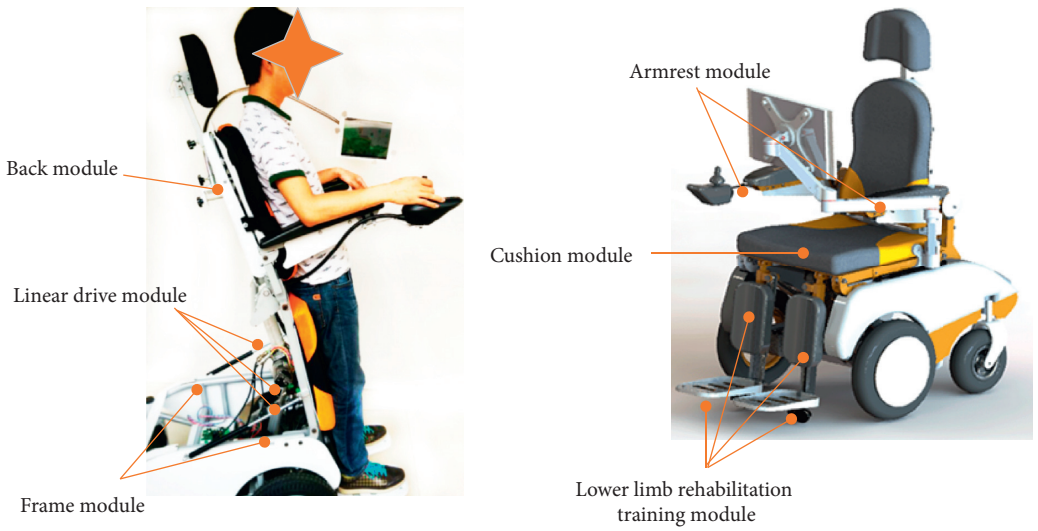


Figure 1: System design of ReChair.

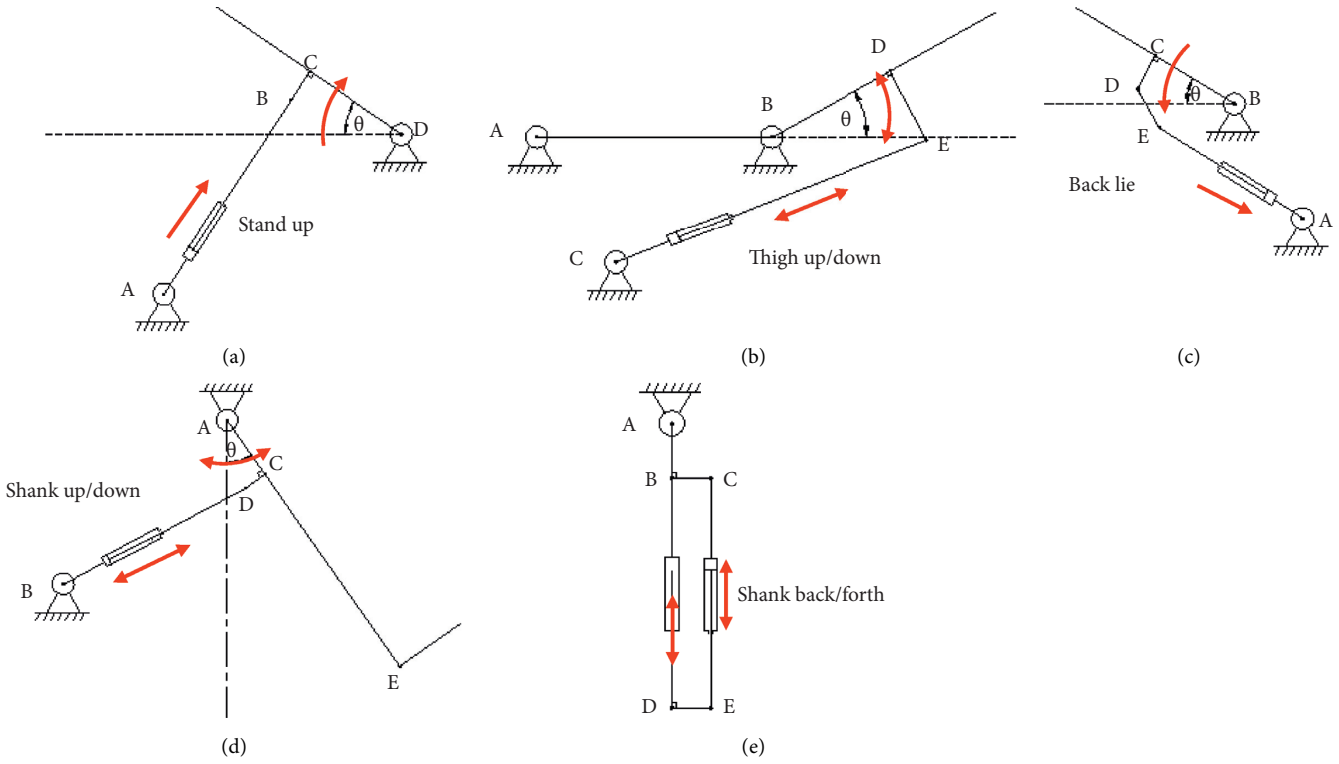


Figure 2: Motion mechanism: (a) stand up; (b) thigh up/down; (c) back lie; (d) shank up/down; (e) shank back/forth.

The sit to lie procedure and horizontal angular displacement of seat, back, and leg support are shown in Figure 4. The same simulation method is used with sit to stand posture transformation. The driving links are lie motor and shank up/down motor. The trajectories of the key components are smooth. When the displacement of the lie motor reaches the limit position, the back is exactly at the horizontal position and no hypokinesis or interference will occur. The sit to stand and sit to lie simulation results show that the multiposture transformation can be realized with ReChair.

The lower limb rehabilitation training procedure during sitting and the trajectory of the unique footplate are shown in Figure 5. The whole procedure is divided into two process: the up and forward movement of the footplate and the down and back movement of the footplate. The footplates are made

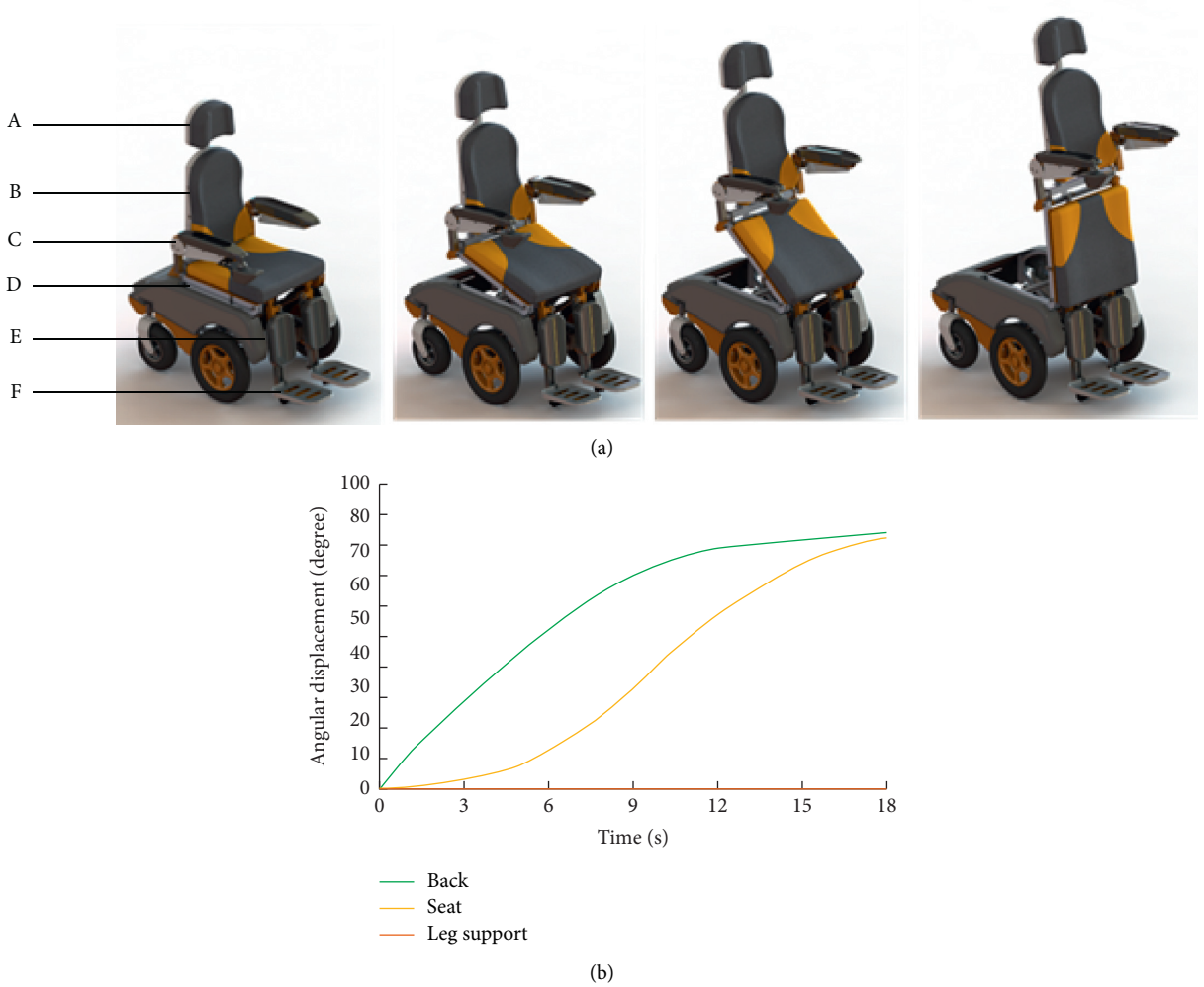


Figure 3: Sit to stand simulation. (a) Sit to stand procedure. A, B, C, D, E, and F are monitoring points. (b) The horizontal angular displacement of seat, back, and leg support.

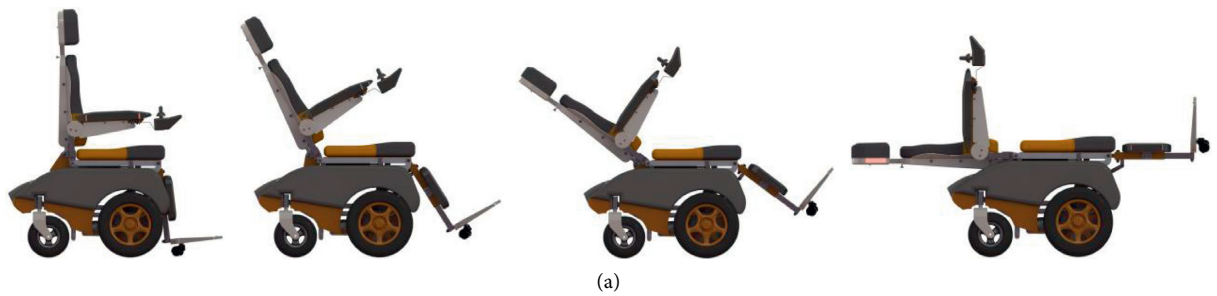
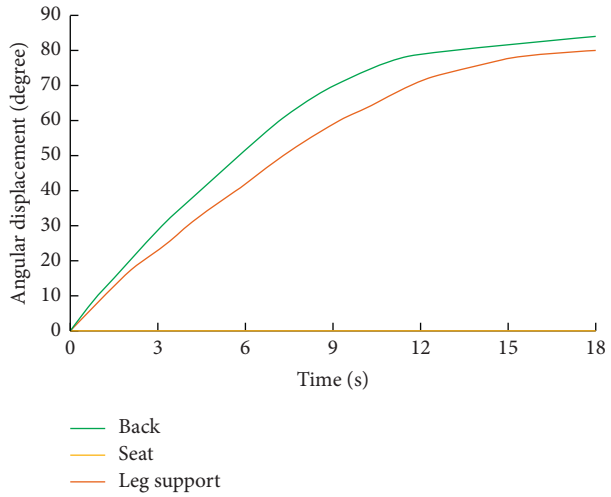


Figure 4: Continued.



(b)

Figure 4: Sit to lie simulation. (a) Sit to lie procedure. (b) The horizontal angular displacement of seat, back, and leg support.

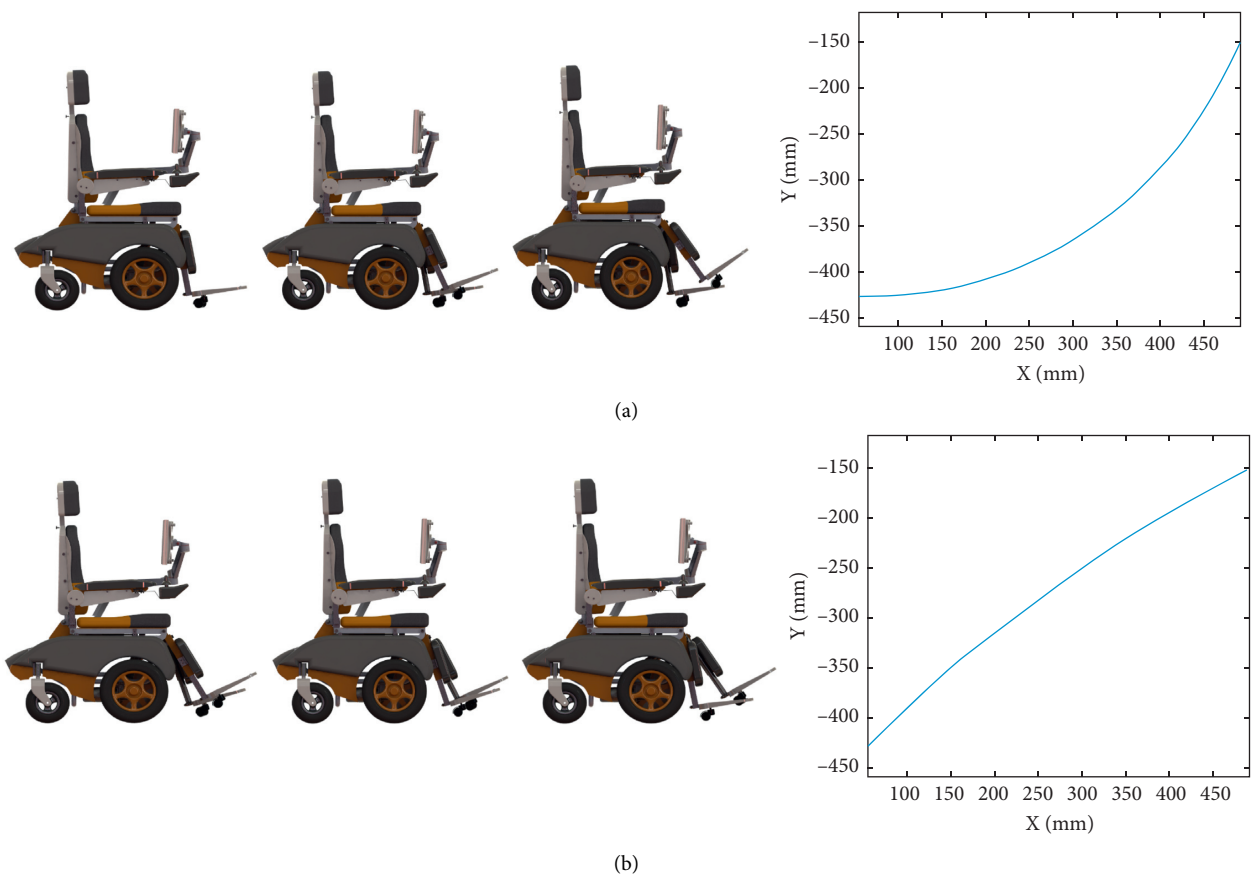


Figure 5: The lower limb rehabilitation training procedure during sit and trajectory of the unique footplate. (a) Footplate up and forward procedure and trajectory. (b) Footplate down and back procedure and trajectory.

of aluminum alloy in this work. The laminated composite structures have shown good performance in dynamic conditions with random vibrations. Saman et al. proposed an accurate finite element model for N-layer MR-laminated beams using a layerwise theory [28] and conducted experimental works on dynamic behavior of laminated composite beam incorporated with magneto-rheological fluid under random excitation [29]. In view of the excellent

characteristics of laminated composite structure, we will optimize the footplate by laminated composite materials in the following work. The footplates of both sides move in opposite directions during the same process. The passive training is similar to circular reciprocating motion. When the speed of the motion is changed, the motion trajectory of the footplate is different as well. Thus, the trajectory planning of the passive lower limb rehabilitation training is realized. The training time and intensity can be set by the nursing personnel or the patients themselves.

2.4. Control System Implementation. Commercially available controllers of the powered wheelchair include hand joystick, sip-n-puff, chin joystick, and head joystick [30]. The input methods including voice, touch, computer vision, accelerometer, and EEG become popular in the smart wheelchair recently [31]. Mousa et al. proposed a smart controller of a wheel chair mobile robot using particle swarm optimization proportional controller (PSO-P) to tune the proportional controller's gains for each axis [32], Rabhi et al. presented a new smart joystick for the electric wheelchair with position controller [33], and Ruzaij et al. described the design and implementation of three modes of operation for a voice controller in the wheelchair. The sound-dependent mode with the dynamic time warping algorithm, the sound-dependent mode with the hidden Markov model algorithm, and the sound-independent mode with the text to sound-independent algorithm were compared [34]. The control system of ReChair uses the idea of modularization and is shown in Figure 6. It is divided into four modules that are the main controller, travelling controller, posture controller, and signal processor.

The main controller is the core part of the ReChair control system. The instructions from the upper computer (mobile phone or smart tablet) and control panel (joystick or button) are received and processed by the main controller. Then, corresponding tasks and data are distributed to each subcontroller. The mode switch of each submodule is adjusted by the main controller to ensure normal function of the control system. Important data or information are fed back to the upper computer via Bluetooth and network communication system. The core chip control circuit is based on STM32F103RCT6 (STMicroelectronics, Switzerland). The STM32F103 incorporates the high-performance ARM® Cortex®-M3 32-bit RISC core operating at a 72 MHz frequency, high-speed embedded memories, and an extensive range of enhanced I/O and peripherals connected to two APB buses. It can satisfy the current and speed control of each motor, the communication between each module, and the acquisition and processing of various sensor signals. The power supplies of the main control module are 5 V and 3.3 V. The battery provides 24 V dc power supply, so voltage regulator chip is used to reduce the voltage to 5 V and 3.3 V. The 5 V DC power supply is obtained through DC-DC chip MP2303 (Monolithic Power Systems, American) and LDO chip LM1117-5.5 V (National Instruments, American). The 3.3 V DC power supply is generated by the 5V power supply through LM1117-3.3 V.

The wireless module in the main controller is responsible for the transmission of data communication and instruction. The chip SI4432 (Silicon Labs, American) is used to design the wireless module. Considering the practical application requirements, the 433 Mhz frequency band is chosen as the operating frequency of the wireless communication module. The module uses SPI communication interface, which is connected to the SPI2 of the main control chip on the main board.

The voice recognition module is embedded in the main controller to achieve convenient interaction between the user and the machine. The sitting, standing, and lying posture transformation can be implemented using voice control. Voice recognition can be divided into speaker-dependent and speaker-independent recognition [35]. Speaker-dependent recognition refers to the recognition of a specific person's voice and is used in the voice module of this system to ensure safety, while speaker-independent recognition refers to the recognition of anyone's voice. The core component used to design the voice recognition module is the NLP-5X (Sensory, American) voice recognition processor. To prevent loss of information during power outages, the EEPROM (electrically erasable programmable read only memory) is used to store speaker-independent templates. The EEPROM chip used for the speech template storage module is 24LC128 (MICROCHIP, America). To obtain the speech template, the voice recognition module collects and processes the speech signals through the microphone and sends them to the processor NLP-5X which then perform software coding of the recorded voice. The processed signal communicates with the MCU (Microcontroller Unit) module and informs it to make corresponding commands. The function of the loudspeaker is to make a sound cue during training and usage.

The joystick is a commonly used human-computer interface tool. Moving the joystick into different positions indicates different output voltage signals, which can easily be captured by electronic components. Thus, the input position signals of the operator can be collected. Small 9000SERIES joystick (APEM, France) with two axis is used in ReChair. It has two potentiometers that can output two mutually perpendicular voltage signals. These two channel voltage signals are connected to the main controller for 12-bit high-precision ADC conversion. The main controller sends the two analog voltage signals and the user-defined wheelchair speed level to the travelling controller through a CAN bus. First, the travelling controller normalizes the position coordinates of the joystick and the wheelchair, and then it calculates the PWM (pulse width modulation) duty cycle of the travelling control signals and the direction of the left and right motors. The rotation direction and speed of the two motors are used to realize the forward, backward, and steering of the wheelchair.

The multiposture transformation module is mainly composed of posture controller, posture driver, and linear actuator. The HT-KC35 (GEMING, China) linear actuator is used as power source for the multiposture transformation. This module is also equipped with three limit switches, two of them are installed at the positive and negative sides of the

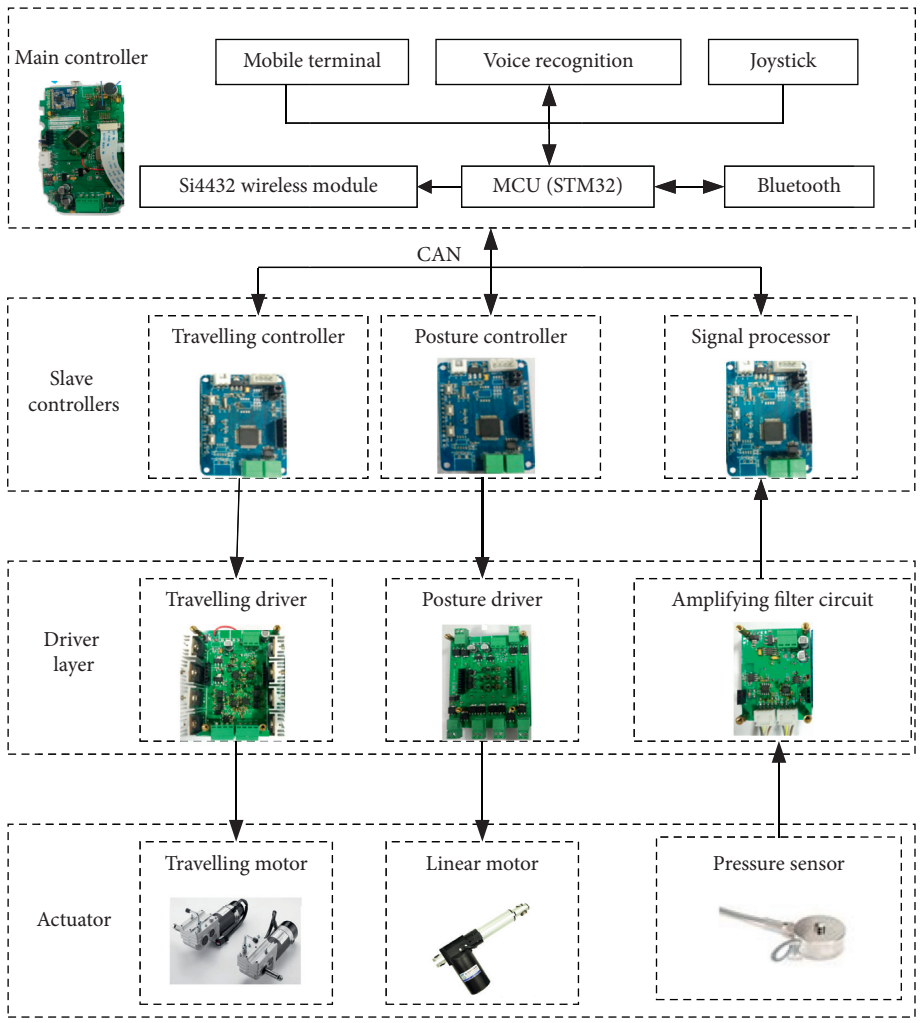


Figure 6: The control system of ReChair.

junction of the back and seat and the third one is installed at the junction of the seat and frame. Although the linear actuator automatically stops when it runs to stroke limit, the controller does not disconnect the control signal then. On the one hand, security risks are hard to avoid, but on the other hand, thermal loss will occur when the MOS tube is alternately switched on and off for a long time. The limit switch does not only ensure safety and save energy but it also informs the controller whether the current posture transformation process is completed.

The main function of the sensor module is the signal acquisition and the processing of the pressure sensors, which is mainly composed of the signal processor, an amplifying filter circuit, and pressure sensors. Two JHBM-100 kg (HBM, Germany) pressure sensors are installed on the left and right footplates of the intelligent rehabilitation wheelchair. In the standing mode, patients can change their center of gravity to control the left and right movements of the game object in order to complete the training game. The pressure sensors detect the change of the center of gravity and convert pressure signals into electrical ones. The deviation side pressure of the center of gravity will increase, resulting in an increase in the output voltage. Lower pressure leads to smaller output voltage of the pressure sensor. The output voltages of the pressure sensors on both sides are amplified by a differential amplification circuit. Then, the signals are used as input to the signal processor after ADC. In the game, the bird will move to the side where the output voltage increases, namely, to the shift side of center of gravity. The greater the voltage difference is, the faster the bird moves.

3. Results and Discussion

3.1. Voice Control System Evaluation. ReChair provides a voice interactive interface so that the posture transformation and lower limb rehabilitation training can be controlled by voice. Seven voice commands (1: sit up, 2: lie down, 3: stand up, 4: sit down, 5: passive training, 6: balance training, and 7: stop) are used as control signals. The user says the control command and the wheelchair controller execute the corresponding action as a success, while the others are recorded as failure. The voice control system is tested by one male and one female users in different environments. First, voice

command training is conducted in quiet environment (20–30 DB). Each command training is conducted twice and stored. Next, two subjects repeat the seven commands in Mandarin, English, and Cantonese 20 times each in quiet environment (20–30 DB) as well. Finally, the above-mentioned process is repeated in a relatively quiet environment (30–40 DB) and a noisy environment (50–60 DB). The success recognition times and rates as well as the mean values are as shown in Table 1. The results indicate that the average success recognition rate can reach 91% to 96% in a quiet environment (20–30 DB), no matter which language is used or what the sex of the subject is. The average success recognition rate is 81% to 84% in a relatively quiet environment (30–40 DB). The range of 20–40 DB is close to home environment. The success recognition rate shows that the voice control system is suitable for control in home environment. While the recognition rate decreases to 49% to 59% during noisy environment (50–60 DB), the results indicate that the voice control system is not suitable for noisy environment. There is little difference in the success rate of different languages during the same environment. Language difference has little effect on the voice control system.

3.2. Sensorless Speed Detection Method Test of Travelling Motor. The speed closed-loop control method is generally used to improve the static and dynamic performance of the DC motor speed control system. The premise of the speed closed loop is the speed detection. Commonly used speed detection methods rely on centrifugal, optical digital, and flash tachometers. Although the accuracy of the sensor method is high, they need to be in a close friction with the rotating part of the motor in the travelling process or fixed in a special position. The installation of the speed detection device increases the structural complexity and the mechanical assembly difficulty of the wheelchair. This work proposes a sensorless speed detection method. When the brush DC motor is working, the DC provided by the H-bridge drive loop is introduced into the armature commutator, and the pulsating current is generated in the drive loop. The pulsation frequency is related to the speed of the DC motor. As long as the frequency information of the pulsating current is collected, the actual speed can be estimated. The pulsating current is collected by the current transformer in series to the circuit to output the pulsating waveform of voltage. The speed of the brush motor is also related to its own related parameters, such as the number of commutator segment, the pairs of magnetic poles, and the parity of the number of the commutator segment [36]. The relation is expressed as

$$f_n = \frac{ckpn}{60}, \tag{3}$$

where f_n indicates the current pulsation frequency, n is the speed of the motor, k is the number of the commutator segment, p indicates the pairs of magnetic poles, and c is the coefficient related to the parity of k such that $c = 1$ when k is an even number and $c = 2$ when k is an odd number. The number of the commutator segment, the pairs of magnetic poles, and the coefficient are all constant values for a specific brush DC motor. Therefore, the speed of the brush DC motor n is proportional to the current pulsation frequency.

$$n = \frac{60 f_n}{ckp}. \tag{4}$$

Considering the reduction ratio i, the wheel speed detected by the sensorless method n_w is

$$n_w = \frac{n}{i} = \frac{60 f_n}{ickp}. \tag{5}$$

To verify the sensorless speed detection method, two travelling motors of the electric wheelchair are selected as experimental objects. The speed measured by the sensorless speed detection circuit is transmitted to the PC using the Bluetooth module of the travelling controller (Figure 7).

For the travelling motor, the values are $p = 8$, $k = 59$, $c = 2$, and $i = 32$. The laser tachometer commonly used in the market is used to measure the actual wheel speed. PWM is used to control the motor speed, and the driving signal frequency is set to 20 KHz. Thirteen sets of duty ratio are given to travelling motors, and the frequency of commutating current is measured. The wheel speeds measured by the sensorless speed detection method and laser tachometer are shown in Table 2. When the PWM is below 15%, such as 10% and 5%, the current is so small that the motor torque cannot overcome the internal friction torque of the motor. The wheelchair stays still, and the speed is zero. When the PWM is enough to drive the wheelchair, the results indicate that the wheel speeds measured by the sensorless method are close to the laser tachometer displayed values. The procreant reasons of plausible error are the interference of the electrical spark generated by the motor on the measured waveform and the influence of distance with the laser tachometer on measuring accuracy. The higher the running speed of brush DC motor, the greater the measurement error. This work investigated the speed detection of the wheelchair-driving motor, whose maximum speed is limited and not suitable for high-speed operations, which meet the requirement.

3.3. Lower Limb Rehabilitation Training Test. The study was approved by our institutional review board, and all subjects have signed informed consent. ReChair includes two training modes: passive training with constraint trajectory and balance training combined with games. Two persons including a healthy person and a patient are involved in the lower limb rehabilitation training test (Figure 8).

Before passive training, the rehabilitation doctor logs into the personal center on the tablet application and makes a rehabilitation training plan. The plan includes the following parameters: training time, maximum range of motion, and training speed. The training plan is synchronized with the wheelchair, and users can then carry out lower limb rehabilitation training of pedaling gait on the wheelchair. The experimental procedure runs as follows:

Table 1: Experimental results of the voice control system.

Subject	Env. (DB)	Commands language		1	2	3	4	5	6	7	Mean
Male	20–30	Man.	Suc. times	19	18	20	19	20	18	19	19
			Suc. rate	95%	90%	100%	95%	100%	90%	95%	95%
		Eng.	Suc. times	18	20	19	20	18	19	19	19
			Suc. rate	90%	100%	95%	100%	90%	95%	95%	95%
		Can.	Suc. times	20	19	19	20	18	19	19	19
			Suc. rate	100%	95%	95%	100%	90%	95%	95%	95%
	30–40	Man.	Suc. times	18	16	15	17	16	17	17	17
			Suc. rate	90%	80%	75%	85%	80%	85%	85%	83%
		Eng.	Suc. times	17	17	16	16	15	17	18	17
			Suc. rate	85%	85%	80%	80%	75%	85%	90%	83%
		Can.	Suc. times	16	16	18	15	16	16	17	16
			Suc. rate	80%	80%	90%	75%	80%	80%	85%	81%
	50–60	Man.	Suc. times	10	12	10	10	11	10	10	10
			Suc. rate	50%	60%	50%	50%	55%	50%	50%	52%
		Eng.	Suc. times	9	11	10	9	10	10	10	10
			Suc. rate	45%	55%	50%	45%	50%	50%	50%	49%
		Can.	Suc. times	11	11	10	9	10	10	11	10
			Suc. rate	55%	55%	50%	45%	50%	50%	55%	51%
Female	20–30	Man.	Suc. times	20	20	19	19	19	19	19	19
			Suc. rate	100%	100%	95%	95%	95%	95%	95%	96%
		Eng.	Suc. times	18	18	18	19	19	18	18	18
			Suc. rate	90%	90%	90%	95%	95%	90%	90%	91%
		Can.	Suc. times	18	17	19	19	19	18	18	18
			Suc. rate	90%	85%	95%	95%	95%	90%	90%	91%
	30–40	Man.	Suc. times	16	16	18	17	17	17	17	17
			Suc. rate	80%	80%	90%	85%	85%	85%	85%	84%
		Eng.	Suc. tmes	16	16	16	17	17	16	16	16
			Suc. rate	80%	80%	80%	85%	85%	80%	80%	81%
		Can.	Suc. times	17	16	16	17	17	16	16	16
			Suc. rate	85%	80%	80%	85%	85%	80%	80%	82%
	50–60	Man.	Suc. times	12	13	11	11	11	12	12	12
			Suc. rate	60%	65%	55%	55%	55%	60%	60%	59%
		Eng.	Suc. times	13	12	12	12	9	10	10	11
			Suc. rate	65%	60%	60%	60%	45%	50%	50%	56%
		Can.	Suc. times	11	10	10	12	11	11	11	11
			Suc. .rate	55%	50%	50%	60%	55%	55%	55%	54%

Man.: Mandarin; Eng.: English; Can.: Cantonese; Suc.: success; 1: sit up; 2: lie down; 3: stand up; 4: sit down; 5: passive training; 6: balance training; 7: stop.

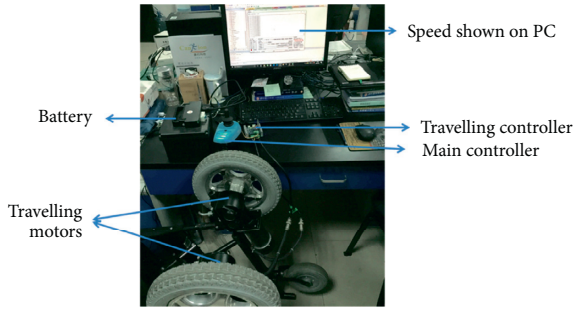


Figure 7: Sensorless speed detection method test of the travelling motor.

(1) Adjust the wheelchair to the sitting position and set the level of the motion range. There are six levels of motion range: the first test is set to level 1 which is the least range of motion, while level 6 is the largest one but is still within the normal motion range. The training time is set to 5 minutes, and the training speed is fixed at moderate level.

(2) Start the training and measure the maximum angles of θ_1 and θ_2.

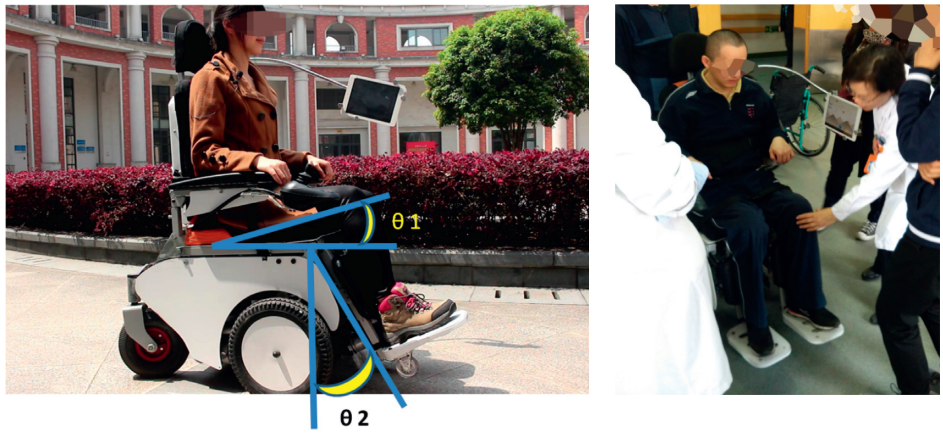
(3) Take a 10-minute break. Increase the range of motion level by one and repeat steps (1) and (2).

(4) Stop the experiment when the subject feels slightly sweatful.

The results are shown in Table 3. The safety must be placed first during rehabilitation training. If the angles range of θ_1 and θ_2 were too low or too large, the training may cause secondary injury to the patients. For θ_1, the maximum thigh up angle was set to be 20°. For θ_2, the maximum shank up angle was set to be 58°. The maximum angle had room for improvement in mechanical design. However, it was a safe threshold for rehabilitation training. The healthy person can complete passive training with all motion levels while the patient cannot complete level 6. Although the motion

Table 2: The wheel speeds measured by the sensorless speed detection method and laser tachometer.

PWM (%)	f_n (Hz)		Estimation speed (rad/min)		Laser tachometer display (rad/min)		Error (%)	
Wheel	Left	Right	Left	Right	Left	Right	Left	Right
5	0	0	0	0	0	0	\	
10	0	0	0	0	0	0		\
15	61	58	7.27	6.91	7.5	7.1	3.06	2.67
20	82	84	9.77	10.01	9.5	9.7	2.84	2.77
30	128	126	15.25	15.02	14.7	15.5	3.74	3.09
40	180	183	21.45	21.80	22.3	21.1	3.83	3.32
50	228	224	27.17	26.69	26.2	27.7	3.71	3.65
60	270	273	32.18	32.53	33.5	33.8	3.94	3.76
70	317	322	38.25	38.37	39.9	36.8	4.13	4.27
80	378	374	45.05	44.57	47.1	46.7	4.35	4.56
90	417	424	49.69	50.53	52.2	48.1	4.81	4.79
95	447	453	53.27	53.99	56.3	56.9	5.38	5.64
98	488	497	58.16	59.23	62.3	63.5	6.65	6.72



(a) (b)

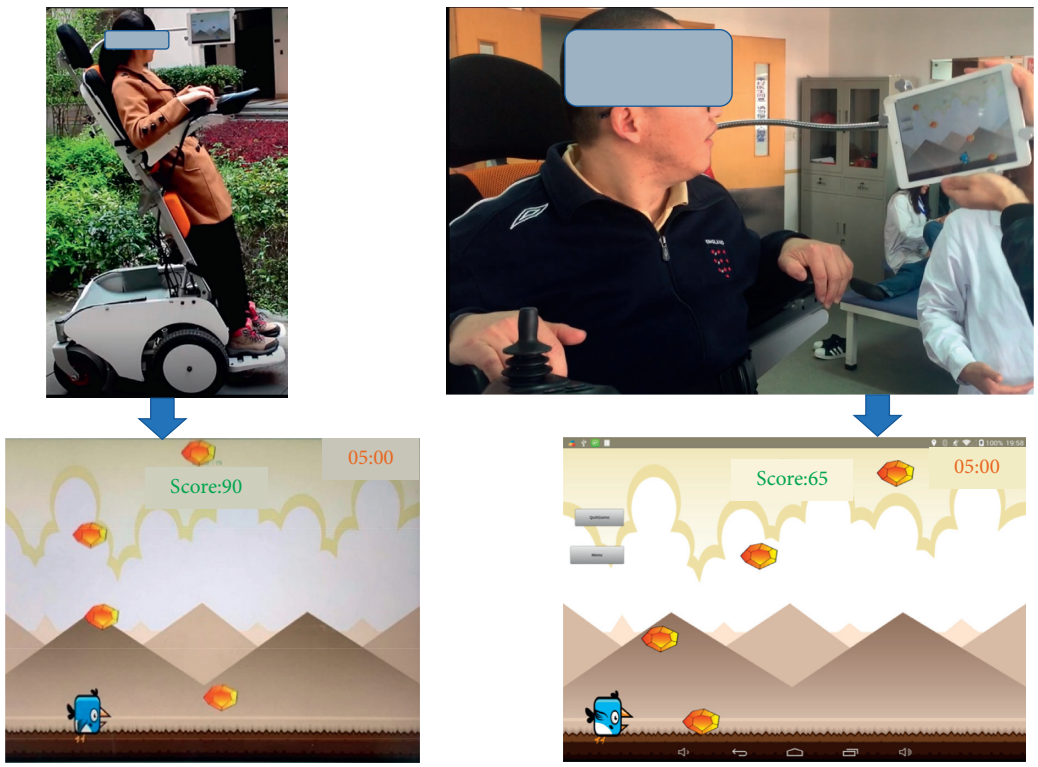
Figure 8: Passive lower limb rehabilitation training. (a) Normal person where θ_1 is the range of hip joint and θ_2 is the range of knee. (b) Patient.

Table 3: The passive training results of normal person and patient.

Subject	Motion level	Complete	Training time (minute)	θ_1(degree)	θ_2(degree)
Normal person	1	Yes	5	10	30
	2	Yes		12	35
	3	Yes		14	41
	4	Yes		16	45
	5	Yes		18	53
	6	Yes		20	58
Patient	1	Yes	5	11	32
	2	Yes		13	36
	3	Yes		15	43
	4	Yes		18	47
	5	Yes		20	55
	6	No		\	\

amplitude of level 6 is too large for the patient, experimental results show that the wheelchair can help users to do passive training, and the range of motion can be adjusted appropriately during the training process.

Balance training is conducted in standing posture while playing the game. The user changes the center of gravity to control the left and right movement of the bird. In the game, the bird will move to the side where the output voltage increases, namely, to the shift side of the center of gravity. The greater the voltage difference is, the faster the bird will move. The game interface is shown in Figure 9. The healthy person gets the high score indicating that the balance



(a) (b)

Figure 9: Balance training combined with game. (a) The score obtained by the normal person. (b) The score obtained by the patient.

training game is easy to operate, while the patient gets a lower score during the training time of five minutes. The result indicates that the balance control ability can still be significantly improved. The patient gets higher score with more training, and this point can be as a quantified guideline for rehabilitation. The game and pressure signal processing module are beneficial and could be an important part of rehabilitation training.

4. Discussion

In this work, we proposed a multifunctional wheelchair (ReChair) that seamlessly integrates the mobility, gait training, and multiposture transformation for daily life conditions. The voice, button, and mobile terminal (mobile phone or smart tablet) control are used for driving and multiposture transformation of ReChair. To evaluate the system, we conducted kinematic simulation and pilot test of ReChair. Results showed that the multiposture transformation by voice control, sensorless speed detection method of the travelling motor, and passive and active training are realized as expected.

Previous research presented lower limb rehabilitation training as mostly conducted by BWSTDs in hospitals. The process of moving a patient from the wheelchair to BWSTD is cumbersome and takes about twenty minutes. This step is skipped by ReChair as the lower limb rehabilitation training is conducted directly on the wheelchair. ReChair reduces the work intensity of the nursing staff and is time saving. ReChair can also be used at home. The low cost of ReChair compared with BWSTDs is more affordable to the patients. Additionally, ReChair reduces the patient's tension on gait rehabilitation machine. Compared with LLE for rewalk [3, 7–13], ReChair focuses on restand and remobility. The multiposture transformation avoids the complications caused by lying down for a long time. The mobility of ReChair meets the travel demand of the patient. It is easier and safer than LLE in daily use. Compared with the wheelchair integrating exercise/rehabilitation system in [25], ReChair includes alternative control modes that make it more widely applicable.

Some extensions and improvement such as smart home module can be integrated into ReChair. Electric control door, lights, and air conditioner in home environment can be combined with ReChair to form a local area network (LAN) system of things. It provides interactive interfaces such as voice recognition, touch terminal, and control panel for people with mobility disabilities so that they can control various electrical equipment at home wirelessly by ReChair. Future goals are to expand the effect of long time use of ReChair on noncomplete SCI and other neurological causes of gait dysfunction. The limitations of this study include the relatively small sample size, the limited range of lower limb motion, and the lack of postintervention follow-up.

5. Conclusions

Restand, mobility, and lower limb rehabilitation training are core requirements of patients with lower limb motor

dysfunction. In this study, we proposed a multifunctional wheelchair ReChair integrating the mobility, gait training, and multiposture transformation in daily life conditions. The whole design concept and detailed specifications of each module were described. The kinematic simulation of multiposture transformation and passive lower limb rehabilitation training were conducted to quantitatively verify the motion capability of ReChair. The voice, button, and mobile terminal (mobile phone or smart tablet) control were used for driving and multiposture transformation of ReChair. The success recognition rate of voice control evaluation shows that the voice control system is suitable for home environment. The sensorless speed detection test results indicate that the wheel speeds measured by the sensorless method are close to the laser tachometer display. The passive and balance training results show that the lower limb rehabilitation training in daily life by ReChair is convenient. In future work, we plan to add a smart home module to ReChair and recruit paraplegic patients for long time rehabilitation training to verify the effectiveness of ReChair.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors thank Shuang Chen, Lulu Wang, and Jie Hu for their assistance of the preparation and participation in the experimental evaluation. This work was supported by the National Natural Science Foundation of China (62003327), Guangdong Basic and Applied Basic Research Foundation (2019A1515110576), and Shanghai Foundation for Development and Technology, China (18441907300).

References

[1] United Nations Department of Economic and Social Affairs, *World Population Prospects 2019: Highlights*, New York, NY, USA, 2019.

[2] W. G. Members, L. Véronique, Roger, A. S. Go et al., "Heart disease and stroke statistics—2012 update A report from the American heart association"" *Circulation*, vol. 127, no. 1, pp. 143–152, 2010.

[3] A. Kozlowski, T. Bryce, and M. Dijkers, "Time and effort required by persons with spinal cord injury to learn to use a powered exoskeleton for assisted walking," *Topics in Spinal Cord Injury Rehabilitation*, vol. 21, no. 2, pp. 110–121, 2015.

[4] J. Wu, J. Gao, R. Song, R. Li, Y. Li, and L. Jiang, "The design and control of a 3DOF lower limb rehabilitation robot," *Mechatronics*, vol. 33, pp. 13–22, 2016.

[5] M. A. A. Erhan Akdoğan, "The design and control of a therapeutic exercise robot for lower limb rehabilitation: Physiotherabot," *Mechatronics*, vol. 21, no. 3, pp. 509–522, 2011.

[6] W. Meng, Q. Liu, Z. Zhou, Q. Ai, B. Sheng, and S. Xie, "Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation," *Mechatronics*, vol. 31, pp. 132–145, 2015.

[7] X. Wu, D.-X. Liu, M. Liu, C. Chen, and H. Guo, "Individualized gait pattern generation for sharing lower limb exoskeleton robot," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 4, pp. 1459–1470, 2018.

[8] W. Cao, C. Chen, H. Hu, K. Fang, and X. Wu, "Effect of hip assistance modes on metabolic cost of walking with a soft exoskeleton," *IEEE Transactions on Automation Science and Engineering*, pp. 1–11, 2020.

[9] C. Wang, X. Wu, and Y. Ma, "Implementation of a brain-computer interface on a lower-limb exoskeleton," *IEEE Access*, vol. 6, pp. 38524–38534, 2018.

[10] A. Esquenazi, M. Talaty, A. Packel, and M. Saulino, "The ReWalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury," *American Journal of Physical Medicine & Rehabilitation*, vol. 91, no. 11, pp. 911–921, 2012.

[11] R. J. Farris, H. A. Quintero, and M. Goldfarb, "Preliminary evaluation of a powered lower limb orthosis to aid walking in paraplegic individuals," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 6, pp. 652–659, 2011.

[12] C. Tefertiller, K. Hays, J. Jones et al., "Initial outcomes from a multicenter study utilizing the Indego powered exoskeleton in spinal cord injury," *Topics in Spinal Cord Injury Rehabilitation*, vol. 24, no. 1, pp. 78–85, 2018.

[13] N. Birch, J. Graham, T. Priestley et al., "Results of the first interim analysis of the RAPPERII trial in patients with spinal cord injury: ambulation and functional exercise programs in the REX powered walking aid," *Journal of NeuroEngineering and Rehabilitation*, vol. 14, no. 60, pp. 1–10, 2017.

[14] S. Mohammed, Y. Amirat, and H. Rifai, "Lower-limb movement assistance through wearable robots: state of the art and challenges," *Advanced Robotics*, vol. 26, no. 1-2, pp. 1–22, 2012.

[15] J. Leaman and H. M. La, "A comprehensive review of smart wheelchairs: past, present, and future," *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 4, pp. 486–499, 2017.

[16] J. H. Choi, Y. Chung, and S. Oh, "Motion control of joystick interfaced electric wheelchair for improvement of safety and riding comfort," *Mechatronics*, vol. 59, pp. 104–114, 2019.

[17] G. Feng, L. Busoniu, T.-M. Guerra, and S. Mohammad, "Data-efficient reinforcement learning for energy optimization of power-assisted wheelchairs," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9734–9744, 2019.

[18] R. C. Murray, C. Ophaswongse, and S. K. Agrawal, "Design of a wheelchair robot for active postural support," *Journal of Mechanisms and Robotics-Transactions of the ASME*, vol. 11, no. 2, pp. 1–9, 2019.

[19] T. Williams and M. Scheutz, "The state-of-the-art in autonomous wheelchairs controlled through natural language: a survey," *Robotics and Autonomous Systems*, vol. 96, pp. 171–183, 2017.

[20] A. Erdogan and B. D. Argall, "The effect of robotic wheelchair control paradigm and interface on user performance, effort and preference: an experimental assessment," *Robotics and Autonomous Systems*, vol. 94, pp. 282–297, 2017.

[21] A. Ishizuka, A. Yorozu, and M. Takahashi, "Driving control of a powered wheelchair considering uncertainty of gaze input in an unknown environment," *Applied Sciences*, vol. 8, pp. 1–20, 2018.

[22] J. L. Candiotti, B. J. Daveler, D. C. Kamaraj et al., "A heuristic approach to overcome architectural barriers using a robotic wheelchair," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 9, pp. 1846–1854, 2019.

[23] L.-C. Hsieh and T.-H. Chen, "The innovative design and prototype verification of wheelchair with one degree of freedom to perform lifting and standing functions," *IOP Conference Series: Materials Science and Engineering*, pp. 1–8, 2017.

[24] O. RabreauS. Chevallier et al., "SenseJoy, a pluggable solution for assessing user behavior during powered wheelchair driving tasks," *Journal of NeuroEngineering and Rehabilitation*, vol. 16, pp. 1–14, 2019.

[25] B. Hwang and D. Jeon, "Development and preliminary testing of a novel wheelchair integrated exercise/rehabilitation system," in *Proceedings of the IEEE 13th International Conference on Rehabilitation Robotics (ICORR)*, pp. 1–6, Seattle, USA, June 2013.

[26] W. Cao, H. Yu, X. Wu, S. Li, Q. Meng, and C. Chen, "Voice controlled wheelchair integration rehabilitation training and posture transformation for people with lower limb motor dysfunction," *Technology and Health Care*, pp. 1–6, 2020.

[27] R.L. Norton, *Design of Machinery*, McGraw-Hill, New York, NY, USA, 2014.

[28] S. Momeni, A. Zabihollah, and M. Behzad, "Development of an accurate finite element model for N-layer MR-laminated beams using a layerwise theory," *Mechanics of Advanced Materials and Structures*, vol. 25, no. 13, pp. 1148–1155, 2018.

[29] S. Momeni, A. Zabihollah, and M. Behzad, "Experimental works on dynamic behavior of laminated composite beam incorporated with magneto-rheological fluid under random excitation," in *Proceedings of the 3rd International Conference on Mechatronics and Robotics Engineering*, pp. 156–161, New York, NY, USA, February 2017.

[30] D. Sinyukov, R. Desmond, M. Dickerman, J. Fleming, J. Schaufeld, and T. Padir, "Multi-modal control framework for a semi-autonomous wheelchair using modular sensor designs," *Intelligent Service Robotics*, vol. 7, no. 3, pp. 145–155, 2014.

[31] F. Pasteau, A. Krupa, and M. Babel, "Vision-based assistance for wheelchair navigation along corridors," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 4430–4435, Hong Kong, China, 2014.

[32] G. Mousa, A. Almaddah, and A. Aly, "Design and implementation of wheel chair control system using Particle Swarm algorithm," *Computers, Materials & Continua*, vol. 66, no. 2, pp. 2005–2023, 2021.

[33] Y. Rabhi, M. Mrabet, and F. Fnaiech, "Optimized Joystick Control Interface for Electric Powered Wheelchairs," in *in Proceedings of the 16th International Conference On Sciences And Techniques Of Automatic Control And Computer Engineering (STA)*, pp. 201–206, Monastir, Tunisia, 2015.

[34] M. F. Ruzaij, S. Neubert, N. Stoll, and K. Thurow, "Design and testing of low cost three-modes of operation voice controller for wheelchairs and rehabilitation robotics," in *Proceedings of the 2015 IEEE 9th International Symposium on Intelligent Signal Processing (WISP) Proceedings*, pp. 1–6, Siena, Italy, May 2015.

[35] J. Makhoul and R. Schwartz, "State of the art in continuous speech recognition," *Proceedings of the National Academy of Sciences*, vol. 92, no. 22, pp. 9956–9963, 1995.

[36] J. Huang, J. Huang, D. Chen et al., "Measurement of DC motor torque-speed characteristic based on wavelet analysis," *Electrical Machine Control and Application*, vol. 28, pp. 49–53, 2016.

Hindawi
Complexity
Volume 2020, Article ID 6687816, 21 pages
https://doi.org/10.1155/2020/6687816

Research Article

Mechatronic Modelling of Industrial AGVs: A Complex System Architecture

J. Enrique Sierra-García [1] **and Matilde Santos** [2]

[1]*Department of Electromechanical Engineering, University of Burgos, Burgos 09006, Spain*
[2]*Institute of Knowledge Technology, Complutense University of Madrid, Madrid 28040, Spain*

Correspondence should be addressed to J. Enrique Sierra-García; jesierra@ubu.es

Received 3 November 2020; Revised 26 November 2020; Accepted 8 December 2020; Published 30 December 2020

Academic Editor: Aydin Azizi

Automatic guided vehicles (AGVs) are unmanned transport vehicles widely used in the industry to substitute manned industrial trucks and conveyors. They are now considered to play a key role in the development of the Industry 4.0 due to their temporal and spatial flexibility. However, in order to deal with the AGV as a potential mobile robot with high capacities and certain level of intelligence, it is necessary to develop control-oriented models of these complex and nonlinear systems. In this paper, the modelling of this vehicle as a whole is addressed. It can be considered composed of several interrelated subsystems: control, safety, driving, guiding and localization, power storage, and charging systems. The kinematics equations of a tricycle vehicle are obtained, and a controller is proposed. An extended hybrid automata formalism is used to define the behaviour of the safety and the control systems, as well as their interaction. In addition, the electrical equivalent circuit of the batteries, charger, and the motors is studied. The architecture of the holistic model is presented. Simulation results of the AGV in a workspace scenario validate the model and prove the efficiency of this approach.

1. Introduction

Automatic guided vehicles (AGVs) are unmanned transport vehicles widely used in the industry to substitute manned industrial trucks and conveyors [1]. They have been proved an efficient element in factory workspaces, helping to reduce logistic errors and operative costs.

Many researchers in manufacturing systems recognize the following Industry 4.0 (I4.0) design principles: interoperability, virtualization, decentralization, real-time capability, service orientation, and modularity. All these principles are connected in some way with the operative of the AGVs. For example, AGVs can easily readjust their program ensuring real-time reaction to unexpected changes in the logistic flows of the production plan. They can also operate in a decentralized way, without any central server commanding orders.

On the other hand, several I4.0 key enabling technologies have been identified in the literature to make these principles possible: autonomous robots, additive manufacturing, simulation, system integration, cloud computing, Internet of things, cybersecurity, augmented reality, and big data. AGVs are included into the category of autonomous robots as one of these technologies. In addition, this AGV can be easily interconnected and integrated with other machines and devices in the plant by its input/output subsystem. Moreover, these AGVs generate a large amount of data which are transported by IOT protocols such as MQTT, stored and processed by cloud computing systems, and can be exploited by big data technologies to obtain KPIs and optimize the production.

For all these reasons, they are now considered to play a key role in the development of the Industry 4.0. Even more, the new concept of collaborative mobile robots seeks the convergence of AGVs and collaborative robots [2].

However, this synergy demands to deal with the AGV as a potential mobile robot with high capacities and certain level of intelligence. Thus, the necessity of control-oriented

models of these complex and strong nonlinear systems is clear. Even more, to take these robots to a further step, understanding their behavior and establishing safety requirements are a must. This will allow them to advance from assembly tasks to a genuine human-robot collaboration.

An AGV is a multivariable complex mechatronic system composed of several subsystems and with a strongly coupled dynamics. Each of the main elements of an AGV influences its dynamics. The power storage affects the control performance and the motion of the AGV. In turn, the control influences the power consumption and thus the power storage. An AGV usually works in a space shared with other robots, so these other vehicles or even people are perceived as obstacles by the safety subsystem that modifies the working conditions accordingly.

Even more, AGVs are normally used to automate logistics and production flows. These processes involve managing a fleet of robots traveling on the same scenario. Indeed, it is considered a multilayer problem [3], where several issues have to be addressed: the design of the layouts, the warehouse and the production plant, the storage strategies, the traffic scheduling, algorithms to share the charging stations, path planning, navigation, routes assignation, and so on. To take all of these simultaneous processes into account, a global model is necessary. This model will also allow to improve the efficiency of these vehicles and the production optimization.

In this paper, we address the modelling and simulation of an AGV as a whole. First, an architecture that relates all the subsystems that are part of the AGV is proposed. Then, each of them has been described and mathematically modelled: the control, safety, driving, guiding and localization, power storage, and charging systems. Besides, the kinematics and dynamics equations of the vehicle are developed. They have been implemented for a real industrial complex AGV, a hybrid tricycle-differential one. This holistic approach gives realistic results in the simulations of the vehicle, which allows us to validate the models.

The main contributions of this work can be summarized as follows:

(i) The development of a general and high level of the abstraction model of an AGV. It has been divided into subsystems whose models can be used to implement different types of AGVs.

(ii) The identification of all the interactions between the subsystems and its integration.

(iii) The development of the mathematical model of the dynamics of a hybrid differential-tricycle AGV (this hybrid differential-tricycle AGV has not been previously studied in the literature).

(iv) The mathematical description of the safety subsystem and the control subsystem based on the hybrid automata formalism. This approach is completely original.

(v) The implementation of this general model into a specific model of a real industrial AGV, the Unibot AGV of the company ASTI Mobile Robotics [4].

(vi) The integration of the kinematics, dynamics, and electrical equations in the simulation of this type of AGV.

The rest of the paper is organized as follows. Section 2 presents a brief state of the art on the AGV modelling. In Section 3, the architecture of the AGV is developed. The description and equations of the different subsystems which compose it are presented in Section 4. Simulation results on a working scenario are discussed in Section 5. The document ends with the conclusions and future works.

2. Related Works

The first paper published on the design and operation of AGV systems was the work by Maxwell and Muckstadt, in 1982 [5], where a methodological framework for specifying the operational characteristics of an automatic guided vehicle system was presented. Since then, the research on this topic has not stopped growing. Nevertheless, the works found in the literature mainly tackle the system focusing on individual aspects of the AGV: kinematics, control, charging, and so on, that are not independent. Indeed, the majority focus on its kinematics and control, or the localization algorithms, or the power storage, or the charging system, but they lack a holistic approach of the guided vehicle as a whole system.

There have been several attempts to model the AGV, for example, Veiga et al. compared vehicle models of a tugger-trolleys system with three linked towed. One model is a complete multibody model of the system. The second is a simplified version composed of vertical and handling models for motion in a plane, assuming no planar forces on the trolleys' front wheels and linking the vehicles with multibody joints and body, representing the coupling arms, developed in Modelica language. Results show good agreement between the models but the second one was much faster and provided more stable results [6].

A kinematical and dynamical analysis of a mobile robot belonging to the configuration classical tricycle is shown in [7]. For the kinematical analysis, authors consider a robot capable of locomotion on a surface by the action of wheels mounted on the robot. The dynamics of the nonholonomic mobile robot is obtained based on the Euler–Lagrange methodology. The paper is focused on the analysis of global asymptotic stability of a new position controller for this wheeled mobile robot.

In [8], authors state that the complexity of AGV dynamics and the difficulty of obtaining the actual vehicle dynamic parameters have led them to use two nonmodel-based control approaches. The robot is equipped with a front-centered steerable wheel and fixed parallel rear wheels. They present a comparison of control techniques for robust docking maneuvers of an AGV, specifically, fuzzy control and vector pursuit that do not require a mathematical model of the vehicle dynamics.

Recently, Montazerijouybari et al. proposed a mathematical model based on the kinematics of 2-DOF AGVs with differential driving wheels and caster wheels in order to

predict the behaviour of the robot when the vehicle is carrying heavy loads not well distributed on it.

A trajectory tracking control of an AGV based on sliding mode control with an improved reaching law is proposed in [9]. The AGV in this paper has two driving wheels and two universal wheels. They consider a fully functional AGV composed of the motion mechanism, the sensing system, and the control system. Authors do not consider the rest of the systems of the vehicle. The work designs a double closed-loop control method to control the vehicle to track the given trajectory. With the same control-oriented aim, in [10], the analysis of the kinematics and dynamics of a four-wheel omnidirectional automated guided vehicle is presented.

There are other AGV-related topics that are dealt with in this paper, such as localization and navigation. In many cases, the routes implemented by these vehicles are subject to constant changes and are adapted to the currently assigned task. Thus, the sensors that allow the AGV to follow the path are an important element of the guided vehicle. Although usually the vehicle sensors are not modelled, the most common ones have been studied. Just a couple of examples are as follows. An intelligent path recognition system for vision-guided AGVs, able to tackle noise in the images captured by using a CCD camera, is presented in [11]. The paper addresses some of the most challenging problems for a vision-based AGV running in a complex workspace and improves the accuracy and reliability of vision recognition of guide paths for AGVs by means of artificial intelligence methods. Authors in [12] proposed an interesting method to measure the wheel radius. This radius is not constant, and in the case of automated guided vehicles using odometry, this is a key information to calculate the position of the vehicle. Stetter et al. designed virtual sensors that estimate forces and torques acting on an AGV [13]. However, this AGVs consists of four sprung arms that each dispose of one drive motor. For this system, they obtain a simple description of the AGV dynamics and then develop a set of virtual sensors that enables estimation of the forces and torques. Indeed, authors say that they try to avoid the use of complex tire models. The study is meant for several diagnostic purposes such as fault detection or fault prevention.

Another AGV component, the charging system, has been also the object of study by some authors. For instance, a battery management simulation is developed for evaluating battery-related costs under various AGV operation modes and for designing battery management strategies [14]. Hou et al. developed a novel energy modelling method which allows to calculate and predict the AGV energy consumption [15]. However, in this work,we do not consider energy efficiency either the design of the battery, but the simulation of the electrical performance of the AGV's battery. The charging method is out of the scope of this paper.

Finally, there are several works about the control of a fleet of AGVs. A detailed state of art on the design and control of several AGVs is presented in [16]. In this paper, author discusses the literature concerning the usage of AGVs in manufacturing and other new areas of these vehicle application, namely, distribution, transshipment, and transportation systems. As a conclusion, the author states that the most important differences between the usage of AGVs in manufacturing areas and in other environments can be noticed in the number of AGVs used. Besides, the author claims the necessity of analytical and simulation models which simultaneously address multiple design and control problems. Different vehicle platooning control strategies are presented in [17, 18].

As shown above, most of the studies found in the literature are mainly focused on individual aspects of the AGV design problem: kinematics, control, guiding, energy, and so on. However, these components are not independent; indeed, these subsystems are highly coupled. Therefore, a holistic and complete vision of these automatic vehicles is missing, where the interactions among their various elements are evident in the models used so that the simulation is realistic.

3. Description of the Industrial AGV Architecture

There are different ways to split up an AGV into its different parts. We have considered the following interconnected subsystems that describe the whole behaviour of the vehicle: driving system, guiding and localization system, safety system, motors, power storage and charging, and control system. These elements interact with the working environment. Figure 1 shows the AGV architecture with all the mentioned systems and their relationships.

The control system can be considered the brain of the AGV. Depending on the localization, on the external inputs and the current state, it adjusts the charging system, generates target signals for the driving system, commands the outputs, and adjusts the safety system. The driving system collects the dynamic and kinematic of the AGV motion; it changes the position and orientation of the AGV and modifies the working environment. It also gives information about the current to the control system in order to close the control loop. The driving system is the component that demands more energy from the power system.

The safety system detects obstacles in the safety zone and warns the control system; it can even trigger an emergency stop. The charging system is responsible for recovering the energy of the power storage system. This subsystem is usually placed out of the vehicle when there is not enough space in the AGV. The power storage is in charge of providing energy to the electronic components and the motors.

The guiding and localization system (GLS) manages AGV position and attitude measures, which are used by the control system. Depending on the technologies used to implement the GLS, this information will have greater precision. There are technologies, such as simultaneous localization and mapping (SLAM), which provide information about the absolute coordinates (x, y, angle) of the AGV with high accuracy. Others only provide the deviation respect to a predefined trajectory, either marked on the floor or buried, such as magnetic antennas, optical sensors, and so on, and must be complemented with a localization system such as RFID tag sensors (low accuracy) or UWB devices (medium accuracy) to know the absolute position.

On the other hand, the input/output system allows to introduce information to the AGV from other external sources: machines, operators, and so on, and extract information. Finally, the workspace represents the external environment and everything that can produce any changes to the AGV behaviour.

In this paper, we are going to model a hybrid differential-tricycle AGV. This vehicle has been selected because nowadays is one of the most used AGVs in the industry. An example of this type of AGVs is the Unibot AGV of the company ASTI Mobile Robotics [4], represented in Figure 2.

The Unibot is mainly used to convey loads by pulling carts. This vehicle is robust, simple, flexible, and reliable and well-established in the market vehicles. Though with reduced dimensions, it has great load capacity. It is fully connected in real time.

This industrial tow AGV can be used in whatever industry where medium loads need to be towed. However, it achieves its full potential in enterprises where recurrent flows of pieces and/or products exist. They can be applied in any SMEs and large companies (for further information on enterprise categories see [19]) although it is commonly found in the automotive industry as it is one of the most automatized sectors. Moreover, it is easy to include them in already functioning enterprises as these automatic guided vehicles do not require to modify the infrastructure in order to implement the layout. Indeed, the layout can be usually easily removed and moved to adapt it to changes in the production plant. However, also in companies that are going to start the production, it is possible to design the plant taking into account the layouts of the AGVs to optimize the intralogistics.

In addition, it has many other functionalities such as magnetic guidance/navigation system (surface/buried), online charging system, safety laser for internal environment, safety PLC, and emergency stop button, among others.

This is a complex system because it has two wheels as a differential robot and a tricycle configuration in the front part. Most of the models found in the literature work as differential robots.

4. Modelling of the AGV's Subsystems

4.1. Driving System. There are many types of AGVs regarding the kinematics, but four different models can be typically identified: tricycle, differential, quad, and omnidirectional. Figure 3 shows some standard configurations of the wheels for these AGVs. Depending on the number and type of the wheels, the kinematic constraints and the control variables of the vehicle will vary. The drive wheels provide traction but cannot change its direction. The steer wheels can change the direction but cannot provide traction. The steer and drive wheels provide both things, and they are also called motor-wheels.

All of these wheels are subjected to nonholonomic constraints to prevent wheel lateral slipping. This can disrupt the control of the vehicle and must be avoided. By contrast, the omnidirectional wheels only provide traction, but they

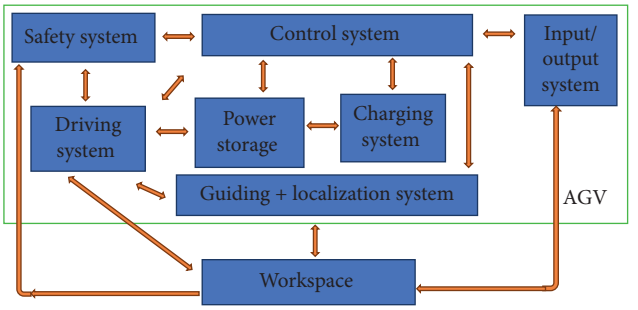


Figure 1: Subsystems AGV architecture.

are not affected by the nonholonomic constraints. In fact, the omni-wheels are equipped with rollers to enable the lateral slipping.

The Unibot AGV [4] is a mixture of tricycle and differential. The traction head is a differential robot; meanwhile, the body is linked with the traction unit by an axle, and it performs as a tricycle. The wheels of the body do not provide traction but support the weight of the vehicle and allow the body to follow the traction head.

As said before, the driving system includes the dynamics and the kinematics of the AGV motion. Therefore, in this section, both the kinematic and dynamic models of the hybrid tricycle-differential AGV are developed. To do it, the kinematic equations of the individual traction tricycle and differential units are combined in a novel way as the hybrid tricycle-differential AGV has not been previously studied in the literature. In addition, the dynamic model of the vehicle has been developed by the authors using the Newton–Euler fundamental equations of the translational and rotational dynamics.

Figure 4 represents the mechanical components of this AGV and the coordinate system (traction head, green square and AGV body, blue rectangle). The center of the wheels of the traction head in the inertial coordinate system is represented by (x_h, y_h), and the position of the center of the wheels of the body is (x_b, y_b), both in m. In this inertial frame, the attitude of the body is Φ_b (rad) and the attitude of the head is Φ_h (rad); the angle between the body and the head is γ (rad). L_b is the distance between (x_b, y_b) and (x_h, y_h) in m, and L_h is the distance between the wheels of the traction unit in m.

The equations of the traction head are given by the kinematic model of a differential robot [20]. That is,

$$\begin{aligned} \dot{x}_h &= \frac{V_l + V_r}{2}\cos(\Phi_h), \\ \dot{y}_h &= \frac{V_l + V_r}{2}\sin(\Phi_h), \\ \dot{\Phi}_h &= \frac{V_r - V_l}{L_h}. \end{aligned} \tag{1}$$

On the other hand, the motion of the body follows the equations of a tricycle:

$$\begin{aligned}\dot{x}_b &= v_h \cos(\gamma)\cos(\Phi_b),\\ \dot{y}_b &= v_h \cos(\gamma)\sin(\Phi_b)\\ \dot{\Phi}_b &= \frac{v_h}{L_b}\sin(\gamma),\end{aligned} \tag{2}$$

where v_h is the longitudinal velocity of the steering drive (m/s), i.e., the traction head speed. These equations (1) and (2) are linked by the following expressions:

$$\begin{aligned}v_h &= \sqrt{\dot{x}_h^2 + \dot{y}_h^2} = \frac{V_l + V_r}{2},\\ \gamma &= \Phi_h - \Phi_b,\\ \gamma_{\min} &\le \gamma \le \gamma_{\max},\end{aligned} \tag{3}$$

where $\gamma_{\min}$ and $\gamma_{\max}$ are the mechanical limits of the angle of the traction head (rad). These mechanical limits are set to avoid damage in the wires between the body and the traction head.

Besides, in order to prevent sliding in the wheels, the following constraints must be met, equation (4) for the rear wheels and equation (5) for the front wheels:

$$\dot{x}_b \sin(\Phi_b) - \dot{x}_b \cos(\Phi_b) = 0, \tag{4}$$

$$\dot{x}_b \sin(\Phi_b + \gamma) - \dot{x}_b \cos(\Phi_b + \gamma) - \dot{\Phi}_b L_b \cos(\Phi_b) = 0, \tag{5}$$

with L_b is the distance between the center of the traction head and the rear axle.

By the translational and rotational Newton–Euler dynamic equations, the following expressions that represent the forces and momentums are obtained:

$$\begin{bmatrix} \frac{m_T \cdot R_h}{2} & \frac{m_T \cdot R_h}{2} \\ \frac{I_h \cdot R_h}{L_h} & -\frac{I_h \cdot R_h}{L_h} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_r \\ \ddot{\theta}_l \end{bmatrix} = \begin{bmatrix} \frac{(M_{er} + M_{el})}{2R_h} - f_{rT} \\ \frac{(M_{er} - M_{el})L_h}{2R_h} - M_{rR} \end{bmatrix},$$

$$\begin{aligned} f_{rT} &= 0.5 \cdot \delta_{\text{air}} \cdot S_{\text{AGV}} \cdot C_{\text{aero}} \cdot \left(v_h^2\right) \cdot \text{sign}(v_h)\\ &\quad + g \cdot m_T \cdot C_{\text{roll}} \cdot \text{sign}(v_h),\\ M_{rR} &= F_{vh} \cdot \dot{\Phi}_h + F_{sh} \cdot \text{sign}\left(\dot{\Phi}_h\right),\\ M_{er} &= M_r - F_{vw} \cdot \dot{\theta}_r - F_{sw} \cdot \text{sign}\left(\dot{\theta}_r\right),\\ M_{el} &= M_l - F_{vw} \cdot \dot{\theta}_l - F_{sw} \cdot \text{sign}\left(\dot{\theta}_l\right),\end{aligned} \tag{6}$$

where m_T (kg) is the total mass of the system, that is, the mass of the AGV, m_{AGV} (kg), plus the mass of the load, m_L (kg); the radius of the wheels in the traction unit is R_h (m); I_h is the rotational inertia of the traction unit (kg m^2); $\ddot{\theta}_r$ and $\ddot{\theta}_l$ are the angular accelerations of the right and left wheel of the traction unit (rad/s^2), respectively;M_{er} and M_{el} are the effective torques in the right and left wheel of the traction unit (Nm); f_{rT} is the translational friction force (N); M_{rR} is the friction torque (Nm); δ_{air} is the density of the air (Kg/m^3); S_{AGV} is the front surface of the air (m^2); C_{aero} is the aerodynamic coefficient; C_{roll} is the rolling coefficient; g is the acceleration of the gravity; F_{vh} and F_{sh} are the viscous and static friction coefficients in the traction unit (Nm and Nms/rad); F_{vw} and F_{sw} are the viscous and static friction coefficients in the wheels (Nm and Nms/rad); M_{er} and M_{el} are the torques produced by the motors (Nm); and sign is the sign function. The effect of the inertia of the wheels has been neglected. It has been supposed that the AGV is working in an indoor environment, and the wind speed is zero.

Figure 2: Unibot AGV.

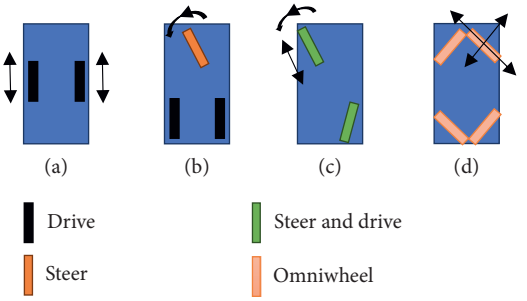


Figure 3: Wheel configurations in AGVs: (a) differential; (b) tricycle; (c) quad; (d) omnidirectional.

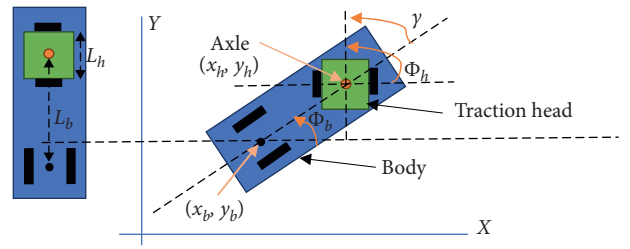


Figure 4: Unibot AGV driving system and the coordinate system.

Moreover, in the simulations, the slippage has been considered. The slippage happens when the traction force exceeds the surface friction force. The traction forces in the front wheels are given by M_{er}/R_h and M_{el}/R_h. Therefore, to avoid the slippage, the following constraints must be fulfilled:

$$\begin{aligned}\frac{M_{\text{er}}}{R_h} &< \frac{g \cdot m_T \mu}{4},\\ \frac{M_{\text{el}}}{R_h} &< \frac{g \cdot m_T \mu}{4},\end{aligned} \tag{7}$$

where μ is the friction coefficient. The number 4 that appears in the denominator means that the weight of the AGV is equally distributed into the four wheels.

4.2. Safety System. In this section, we develop the mathematical model of a general safety subsystem applying a new approach to do this, the hybrid automata formalism. The safety system is in charge of preventing collisions with humans and objects. The safety norms establish that the AGV must stop in the near presence of obstacles. This normally leads to introduce a processor in the safety system to take the control and trigger an emergency stop command when the control system is not operative.

Figure 5 shows the interactions between the driving, the safety, and the control systems. The safety system monitors the velocities of the wheels (v_l, v_r) by encoders. If the current longitudinal velocity is higher than the predefined current safety velocity, v_{safe}, during a configurable period of time, then the safety system fires an emergency stop procedure. This is represented in Figure 5 as the stop signal. Normally, this is carried out by acting directly on the brake of the vehicle or on the motors. The control system is also informed about this anomaly situation by the stop signal.

On the other hand, the control system notifies the safety system about what type of safety zones must be activated at each moment. These safety zones are polygonal areas where there cannot be obstacles. Each zone has a code$\in \mathcal{N}$ and a maximum velocity, $v_{\text{safe}} \in \mathcal{R}$ (m/s), which is calculated considering the deceleration rate of the AGV to ensure that it is able to stop before a collision. The control system knows the location of the AGV and, for instance, if the AGV is towing a trolley, the safety zones must be big enough to cover it. Moreover, if the AGV is going to turn left, the safety zones must be defined on this side of the AGV. If the safety system detects an obstacle inside the safety perimeter, it automatically sends the code of the zone to the control system, with the signal code, and updates the safety velocity.

The AGV safety system is equipped with different types of sensors: mechanical, ultrasounds, infrared, or LIDAR, to detect the obstacles. In the Unibot AGV, the safety system has a PLC and a LIDAR sensor installed in the front of the vehicle. The safety zones can be mathematically defined as a set $Z_i \subseteq \mathcal{R}^4$ of pairs of points relative to the center of the LIDAR $(x_{\text{LID}}, y_{\text{LID}})$. In addition, each safety zone can be classified as warning zone or error zone. Therefore, by the union of all zones, it is possible to conform the set $\text{Zones} \subseteq Z_i \times \mathcal{N} \times \mathcal{R}$.

The LIDAR returns a set of distance measures, P_{DIST}, that are defined in polar coordinates as follows:

$$\begin{aligned} P_{\text{DIST}} \subset (r,\theta) \in (\mathcal{R} < r_{\max}) \times \theta_C|, \\ \forall (r_1,\theta_1), (r_2,\theta_2) \theta_1 = \theta_2 \Rightarrow r_1 = r_2, \\ \theta_C \subset \theta \in \mathcal{R} | \theta = (i-1) \cdot \text{res} + \theta_{\min}, \\ i \in \mathcal{N} \leq N_p. \end{aligned} \tag{8}$$

The size of this set is $N_p = \text{res} \cdot (\theta_{\max} - \theta_{\min})$, where res is the angular resolution, and $\theta_{\min}$ and $\theta_{\max}$ are minimum and maximum angles detected by the LIDAR. This set may be transformed to Cartesian coordinates by the following function:

$$\begin{aligned} f_{xy}\colon P_{\text{DIST}} &\longrightarrow \mathcal{R}^2, \\ P_{XY} = f_{xy}(r,\theta) &= (r\cos(\theta), \rho\sin(\theta)). \end{aligned} \tag{9}$$

(1) The subset of allowed zones, $\text{Zones}_{\text{EN}} \subseteq \text{Zones}$, is obtained according to the type of zone, $\text{type}_{\text{zone}}$, selected by the control system. The safety PLC checks if any of the points in P_{XY} belongs to the set of allowed zones by the function: $f_{\text{zone}}\colon P_{XY} \longrightarrow \text{Zones}_{\text{EN}}$ (9).

$$f_{\text{zone}}(x,y) = \begin{cases} (\varnothing, 0, v_{\text{smax}}), & \forall (z,c,v) \in \text{Zones}_{\text{EN}} \,|\, (x,y) \notin \text{Polygon}(z), \\ (z,c,v) \in \text{Zones}_{\text{EN}} | (x,y) \in \text{Polygon}(z), & \exists (z,c,v) \in \text{Zones}_{\text{EN}} | (x,y) \in \text{Polygon}(z), \end{cases} \tag{10}$$

where the function $\text{Polygon}(z)\colon Z_i \longrightarrow (\mathcal{R},\mathcal{R})^M$ gives the list of points within the perimeter of the safety zone. If there is not any obstacle in an allowed safety zone, the zone given by (10) is $(\varnothing, 0, v_{\text{smax}})$, being v_{smax} (m/s) the AGV maximum velocity. This v_{smax} is usually defined in the datasheet of the vehicle, and it normally goes from 0.8 m/s to 2 m/s.

If more than one zone is obtained by (10), the safety velocity, v_{safe}, is set as the minimum safe velocity. This can be expressed as follows:

$$v_{\text{safe}} = \text{MIN}(v | (z,c,v) = f_{\text{zone}}(x,y), \quad \forall (x,y) \in P_{XY}). \tag{11}$$

The model of the safety controller can be described by the hybrid automata formalism. A hybrid automata is a formal model that combines discrete control graphs, usually called finite state automata, with continuously evolving variables. An extension to the hybrid automata H defined in [21] is a tuple $H = (Q, X, U, Y, \text{Init}, D, \text{Dom}, E, G, R)$, where

(i) **Q** is a set of discrete variables, and Q is countable. Also it is also known as the set of discrete states of the automata.

(ii) **X** is a set of continuous variables, that is, the set of continuous state variables of the automata.

(iii) **U** is a set of input variables.

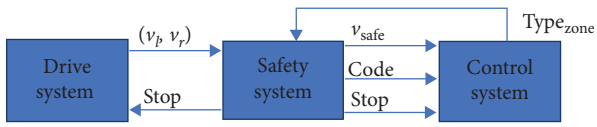


Figure 5: Interactions between driving system, safety system, and the control system.

(iv) $\mathbf{Y}$ is a set of output variables.
(v) $\mathbf{Init} \subseteq Q \times X \times U$ is a set of initial states.
(vi) $\mathbf{D}$: $Q \times X \times U \longrightarrow \text{TX}$ is a vector field of the differential equations.
(vii) $\mathbf{Dom}$: $Q \longrightarrow P(\text{X})$ assigns a domain to each $q \in Q$.
(viii) $\mathbf{E} \subset Q \times Q$ is a collection of discrete transitions.
(ix) $\mathbf{G}$: $E \longrightarrow P(X \times U, X \times U)$ assigns a guard condition to each $e = (q, q') \in E$.
(x) $\mathbf{R}$: $E \times X \times Y \longrightarrow P(X \times Y)$ assigns a reset relation to each $e = (q, q') \in E$ and $x \in X$.

Let $\mathbf{Q} = \{q_1, q_2, q_3, q_4\}$ be the set of discrete variables; $\mathbf{X} = \{t\}$ is the set of continuous variables; $\mathbf{U} = \{v_l, v_r, \text{type}_{zone}, P_{DIST}\}$ is the set of input variables; $\mathbf{Y} = \{\text{stop} \in \{0, 1\}, v_{safe}, \text{code}\}$ is the set of output variables, and $\mathbf{Init} = \{q_1, t = 0, \text{stop} = 0\} \in Q \times X \times U$ is the initial. When the variable stop = 1, the emergency stop is triggered. As expected, the safety controller receives as inputs the velocity of the wheels, the type of allowed zones, and the points measured by the LIDAR; its output is the stop signal, the current safe velocity, and the code of the active safety zone.

The differential equations are given by

$$\mathbf{D}(q_1, x) = \mathbf{D}(q_3, x) = 0, \mathbf{D}(q_2, x) = \mathbf{D}(q_4, x) = 1. \quad (12)$$

The domains are $\mathbf{Dom}(q_1) = \mathbf{Dom}(q_3) = 0$, $\mathbf{Dom}(q_2) = \{t \in \mathscr{R} < t_2\}$, and $\mathbf{Dom}(q_4) = \{t \in \mathscr{R} < t_4\}$. In addition, the discrete transitions are given by

$$\mathbf{E} = \{(q_1, q_2), (q_2, q_1), (q_2, q_3), (q_3, q_4), (q_4, q_3), (q_4, q_1)\}. \quad (13)$$

The guard conditions are the following:

$$\begin{aligned}
\mathbf{G}(q_1, q_2) &= \{(t, v_l, v_r, \text{type}_{zone}, P_{DIST}) \in X \times U \mid \frac{(v_l + v_r)}{2} > v_{safe}(\text{type}_{zone}, P_{DIST})\},\\
\mathbf{G}(q_2, q_1) &= \{(t, v_l, v_r, \text{type}_{zone}, P_{DIST}) \in X \times U \mid \frac{(v_l + v_r)}{2} \leq v_{safe}(\text{type}_{zone}, P_{DIST})\},\\
\mathbf{G}(q_2, q_3) &= \{(t, v_l, v_r, \text{type}_{zone}, P_{DIST}) \in X \times U \mid t \geq t_2\},\\
\mathbf{G}(q_3, q_4) &= \{(t, v_l, v_r, \text{type}_{zone}, P_{DIST}) \in X \times U \mid \frac{(v_l + v_r)}{2} \leq v_{safe}(\text{type}_{zone}, P_{DIST})\},\\
\mathbf{G}(q_4, q_3) &= \{(t, v_l, v_r, \text{type}_{zone}, P_{DIST}) \in X \times U \mid \frac{(v_l + v_r)}{2} > v_{safe}(\text{type}_{zone}, P_{DIST})\},\\
\mathbf{G}(q_4, q_1) &= \{(t, v_l, v_r, \text{type}_{zone}, P_{DIST}) \in X \times U \mid t \geq t_4\},
\end{aligned} \quad (14)$$

where v_{safe} is calculated as a function of type_{zone} and P_{DIST} by equations (9)–(11) and t_2 and t_4 are parameters of the safety controller which can be adjusted by the user.

The reset relations can be expressed as follows:

$$\begin{aligned}
\mathbf{R}(q_1, q_2, x, \text{stop}) &= \{0, \text{stop} = 0\},\\
\mathbf{R}(q_2, q_1, x, \text{stop}) &= \{t, \text{stop} = 0\},
\end{aligned} \quad (15)$$

$$\begin{aligned}
\mathbf{R}(q_2, q_3, x, \text{stop}) &= \{t, \text{stop} = 1\},\\
\mathbf{R}(q_3, q_4, x, \text{stop}) &= \{0, \text{stop} = 1\},
\end{aligned} \quad (16)$$

$$\begin{aligned}
\mathbf{R}(q_4, q_3, x, \text{stop}) &= \{t, \text{stop} = 1\},\\
\mathbf{R}(q_4, q_1, x, \text{stop}) &= \{t, \text{stop} = 0\}.
\end{aligned} \quad (17)$$

The outputs: v_{safe} and code, are calculated in the same way for all discrete states, by equations (10) and (11); thus, they have not been included in (15)–(17).

To summarize, if an obstacle is detected in a warning zone, the maximum allowed speed is reduced. If the obstacle is in an error zone, the maximum speed is automatically set to zero. In both cases, the control system receives a warning, and it must follow the commands of

the safety system. If the safety system detects that the AGV velocity is higher than the allowed speed for a period of time, the safety chain is broken and the vehicle is stopped.

4.3. Guiding and Localization System

The guiding and localization functionalities of the AGV can be implemented by only one device, such as a navigation system based on laser reflectors or the SLAM navigation systems. Both technologies are commonly complemented with the use of an accurate odometry of the vehicle [22]. To combine this information that come from different sensors, a Kalman filter or another similar mathematical tool can be used. However, there are situations, such as slippage produced by an inadequate load distribution or a slippery surface, where the estimation of the speed using only the encoders of the wheels is not possible. In these cases, an inertial measurement unit (IMU) embedded in the vehicle can help to estimate the AGV velocity. However, this solution increases the cost of the AGV, and thus many mobile robots are not equipped with the IMU neither implement a Kalman filter. This is the case of the AGV studied here. As it will be explained below, in this AGV, guiding and localization are decoupled. Guiding (understood as deviation from the expected trajectory without localization) is provided by the magnetic sensor, and the localization is provided by a RFID reader.

The guiding system allows us to know the deviation between the AGV and the path defined in the working environment. There are different types of guiding sensors: optical ones (to follow a painted line), magnetic ones (to follow a magnetic tape), inductive ones (to follow a buried wire), and so on. In the AGV used in the study, the guiding is based on a magnetic sensor.

The localization system detects points of reference placed in the working environment, with a known specific location on the plane. These reference points may be sensed by an optical device (QR codes), a magnetic device (magnetic spots), an electromagnetic device (RFID tags), and so on. This way the AGV can know its own location through references. The accuracy of the localization system is half the distance between two points. In the standard Unibot robot, the localization is RFID-based.

The traction unit of the AGV equipped with the guiding sensor and the magnetic tape on the floor is shown in Figure 6. The guiding sensor gives the error signal $\mathrm{err}_{\mathrm{gui}}$, that is, the deviation respect to the planned trajectory. If the sensing point is at the right of the magnetic tape, the error is considered positive; otherwise, it is negative.

4.4. Control System. The main contribution of this section is the formal description of a general control subsystem of the AGV, as well as the identification of the interactions with the other subsystems. Figure 7 shows the interaction between the subsystems driving and safety, the guiding and location systems, as well as the internal structure of the control system (orange dashed rectangle). The control system has three internal blocks: the control unit, the velocity control, and a block called MIN that calculates the minimum of its inputs.

The control unit has as inputs the localization variable from the GLS and the stop signal from the safety system. Its outputs are the reference of the cruise velocity, v_{ref}, and the type of safety zone, $\mathrm{type}_{\mathrm{zone}}$, associated to the section of the route where the AGV is currently located.

The inputs of the MIN block are v_{ref} and v_{safe} (from the safety system). It selects the minimum of them and calculates the effective velocity that the AGV should have, v_{foll}.

Finally, the velocity control module is in charge of generating the control signals for the motors (u_l, u_r) in order to track the reference longitudinal velocity v_{foll} and to minimize the error signal, $\mathrm{err}_{\mathrm{gui}}$. To do it, it receives the velocity of the wheels (v_l, v_r) from the driving system and the error signal $\mathrm{err}_{\mathrm{gui}}$ from the GLS. Even more, it is notified if the stop signal is triggered, and it also receives the references for the wheel speeds $(\mathrm{man}, u_{\mathrm{ml}}, u_{\mathrm{mr}})$ directly from the user if the manual controller is on. It interacts with the workspace through a set of input/output signals ($\mathrm{IO}_{\mathrm{signals}}$).

The velocity control is carried out by the function $f_{v\mathrm{con}}$, which receives as inputs v_{foll} (m/s) from the MIN block, $\mathrm{err}_{\mathrm{gui}}$ (m) from the GLS, (v_l, v_r) (m/s) from the driving system, $(u_{\mathrm{ml}}, u_{\mathrm{mr}})$ (m/s) from an external control device, and the digital signals $(\mathrm{stop}, \mathrm{man}) \in \{0, 1\}^2$. The stop signal, as it was explained, comes from the safety system. The man signal is generated in an external control device. It is 1 when the manual external controller device is on and 0 otherwise. The output of the $f_{v\mathrm{con}}$ function is the pair of control signals for the motors $(u_l, u_r) \in \{u \in \mathscr{R} | -t1n \leq quh \leq {}_1\}^2$. Thus, this function $f_{v\mathrm{con}}(v_{\mathrm{foll}}, \mathrm{err}_{\mathrm{gui}}, v_l, v_r, u_{\mathrm{ml}}, u_{\mathrm{mr}}, \mathrm{stop}, \mathrm{man})$: $\mathscr{R}^6 \times \{0, 1\}^2 \longrightarrow \{u \in \mathscr{R} | -t1n \leq quh \leq {}_1\}^2$ can be defined as follows:

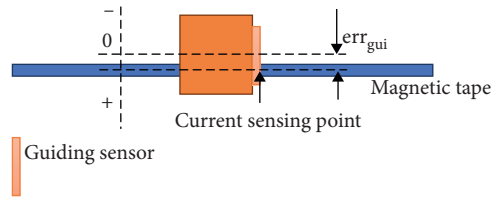


Figure 6: Guiding sensor of the AGV.

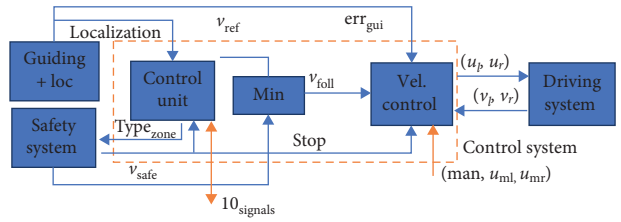


Figure 7: The control system and the interaction with other subsystems.

$$\dot{\Phi}_{h\text{ref}} = \text{PID}\left(\text{err}_{\text{gui}}\right) = K_{\Phi P} \cdot \text{err}_{\text{gui}} + K_{\Phi D} \cdot \dot{\text{err}}_{\text{gui}} + K_{\Phi I} \int \text{err}_{\text{gui}}, \tag{18}$$

$$\begin{aligned} v_{l\text{ref}} &= v_{\text{foll}} - \frac{\left(\dot{\Phi}_{h\text{ref}} \cdot L_h\right)}{2}, \\ v_{r\text{ref}} &= v_{\text{foll}} + \frac{\left(\dot{\Phi}_{h\text{ref}} \cdot L_h\right)}{2}, \end{aligned} \tag{19}$$

$$\left(u_l, u_r\right) = f_{v\text{con}}\left(\ldots\right) = \begin{cases} (0,0), & \text{if stop} = 1, \\ \left(u_{\text{ml}}, u_{\text{mr}}\right), & \text{if stop} = 0 \wedge \text{man} = 1, \\ \left(\text{PI}\left(v_{l\text{ref}}, v_l\right), \text{PI}\left(v_{r\text{ref}}, v_r\right)\right), & \text{if stop} = 0 \wedge \text{man} = 0. \end{cases} \tag{20}$$

As it may be observed in (19), when stop signal is active, the motor targets are set to 0. However, when the manual mode is active, the external velocity control signals $(u_{\text{ml}}, u_{\text{mr}})$ are changed to (u_l, u_r). On the other hand, if the automatic mode is on (last statement in (20)), the control output is generated by a couple of PID controllers, in particular a PI controller per wheel. The targets for the left and right wheels are $v_{l\text{ref}}$ and $v_{r\text{ref}}$, respectively. These targets are calculated by equation (19) using as inputs the longitudinal velocity reference, v_{foll}, and the angular velocity reference, $\dot{\Phi}_{h\text{ref}}$. The angular reference is computed by a PID to compensate the guiding error, err_{gui}.

As explained before, the control unit is in charge of assigning the type of safety zone, $\text{type}_{\text{zone}}$, and a cruise velocity reference, v_{ref}. A lookup table is generated by the user with localization as input and this information.

Likewise with the safety controller, the control unit can be modelled as a hybrid automata. It has three macrostates: automatic, manual, and safety stop. Therefore, the set of discrete states is $\mathbf{Q_{CS}} = \{\text{auto}, \text{manual}, \text{safestop}\}$; the set of continuous variables is $\mathbf{X_{CS}}$, that is, defined by the user according to the actions to be carried out by the AGV. $\mathbf{U_{CS}} = \{\text{Localization}, \text{man}, \text{inp}_{\text{prog}}\}$ is the set of input variables and $\mathbf{Y_{CS}} = \{\text{type}_{\text{zone}}, v_{\text{ref}}, \text{out}_{\text{prog}}\}$, where the inputs inp_{prog} and outputs out_{prog} are configured by the user in the lookup table. The initial state is **Init CS** $= \{\text{auto}, \text{type}_{\text{zone}} = 0, v_{\text{ref}} = 0\} \in Q_{\text{CS}} \times X_{\text{CS}} \times U_{\text{CS}}$.

The set of discrete transitions is given by

$$\mathbf{E_{CS}} = \{(\text{auto}, \text{manual}), (\text{auto}, \text{safestop}), (\text{manual}, \text{auto}), (\text{manual}, \text{safestop}), (\text{safestop}, \text{auto}), (\text{safestop}, \text{manual})\}. \tag{21}$$

The set of guard conditions are given by

$$\begin{aligned}\mathbf{G_{CS}}(\text{auto}, \text{manual}) &= G(\text{safestop}, \text{manual}) = \{\text{man} = 1 \wedge \text{stop} = 0\},\\ \mathbf{G_{CS}}(\text{manual}, \text{auto}) &= G(\text{safestop}, \text{auto}) = \{\text{man} = 0 \wedge \text{stop} = 0\},\\ \mathbf{G_{CS}}(\text{auto}, \text{safestop}) &= G(\text{manual}, \text{safestop}) = \{\text{stop} = 1\}.\end{aligned} \tag{22}$$

The hybrid automata is defined for each macrostate: H_{auto}, H_{man}, and H_{stop} by a tuple $H = (Q, X, U, Y, \text{Init}, D, \text{Dom}, E, G, R)$, as defined before. In this work, we are going to focus on H_{auto} because it has more interactions with the rest of subsystems. When in the automatic macrostate, the set of discrete states is $\mathbf{Q_{CSauto}} = \{w\text{con}_i, \text{march}_i\}\, i \in \mathcal{N} < N_{\text{loc}}$, where N_{loc} is the number of known localizations. In other words, the number of microstates is $2 \cdot N_{\text{loc}}$. The parameter N_{loc} is configured by the user for each application. The discrete variables $w\text{con}_i$ mean "waiting for conditions," and the variables march_i mean "the AGV is in motion." Therefore, for each localization defined in the lookup table, the AGV will be either waiting for conditions or in motion once the conditions are fulfilled.

The set of continuous variables in this macrostate, $\mathbf{X_{CSauto}}$, and its corresponding vector field of differential equations, $\mathbf{D_{CSauto}}$, are defined by the user according to the actions to be carried out by the AGV. The user can associate to each localization_i in the lookup table a set of inputs $\text{inp}_{i\text{prog}}$, a set of outputs $\text{out}_{i_{\text{prog}}}$, a set of conditions Cond_i, a cruise velocity reference $v_{i\text{prog}}$, and a type of safety zone $tsf_{i\text{prog}}$. Thus, the set of inputs in this macrostate is $\mathbf{U_{CSauto}} = \{\text{Localization}\} \cup \text{inp}_{\text{prog}}$, and the set of outputs is $\mathbf{Y_{CSauto}} = \{\text{type}_{\text{zone}}, v_{\text{ref}},\} \cup \text{out}_{\text{prog}}$.

The set of discrete transitions in the automatic mode is as follows:

$$\begin{aligned}\mathbf{E_{CSauto}} = &\{(w\text{con}_i, \text{march}_i)\} i \in \mathcal{N}\\ &< N_{\text{loc}} \cup \{(\text{march}_i, \text{cond}_j)\}\, (i, j) \in \mathfrak{R}_{\text{prog}},\end{aligned} \tag{23}$$

where $\mathfrak{R}_{\text{prog}}$ is a user-defined relationship. This relationship defines the sequence of localization values of an action and so it is necessary to specify a pair of microstates $(\text{march}_i, \text{cond}_i)$. As shown in (23), the jump between two microstates can only be done if its sequence appears in $\mathfrak{R}_{\text{prog}}$.

From the relationship $\mathfrak{R}_{\text{prog}}$, the set of guards is automatically generated as follows:

$$\mathbf{G_{CSauto}}(w\text{con}_i, \text{march}_i) = \{\text{Cond}_i\}, \quad i \in \mathcal{N} < N_{\text{loc}}, \tag{24}$$

$$\mathbf{G_{CSauto}}(\text{march}_i, w\text{con}_j) = \{\text{localization} = j \wedge (i, j) \in \mathfrak{R}_{\text{prog}}\}, \quad (i, j) \in (\mathcal{N}^2 < N_{\text{prog}}, i \in N_{\text{loc}}), \tag{25}$$

where $\{\text{Cond}_i\}\, i \in \mathcal{N} < N_{\text{loc}}$ is a set of conditions. It is possible to observe in (23) and (24) how the AGV in the microstate $w\text{con}_i$ waits for the meeting of condition Cond_i. When they are fulfilled, it switches to the microstate march_i. And vice versa, when it is at march_i state and it reaches a localization defined in $\mathfrak{R}_{\text{prog}}$, it changes to $w\text{con}_j$.

Finally the set of reset relations is given by

$$\mathbf{R_{CSauto}}(w\text{con}_i, \text{march}_i) = \{\text{type}_{\text{zone}} = tsf_{i\text{prog}}, v_{\text{ref}} = v_{i\text{prog}}, \text{out}_{\text{prog}} = \text{out}_{i_{\text{prog}}}\}, \quad i \in \mathcal{N} < N_{\text{loc}}. \tag{26}$$

$$\mathbf{R_{CSauto}}(\text{march}_i, w\text{con}_j) = \{\text{type}_{\text{zone}} = tsf_{j\text{prog}}, v_{\text{ref}} = 0, \text{out}_{\text{prog}} = \text{out}_{j_{\text{prog}}}\}, \quad (i, j) \in \mathfrak{R}_{\text{prog}}. \tag{27}$$

As expressed in (26), when the AGV reaches the microstate march_i, it sets v_{ref} to the value $v_{i\text{prog}}$ stored in the lookup table associated to localization_i.

4.5. Electrical System: Charging, Power Storage, and Motors. The novelty of this section is the integration of the electrical models of the battery, the charger, the motors, and the drives. The most frequently power storage technologies used in AGVs are the lead-based and lithium batteries. More recently, supercaps, electric capacitors with an unusually high power density, are being used for very high charging currents demand. However, still lead batteries are the most widely used because they are much cheaper than the other options. There are several mathematical models of these batteries, each one considering different aspects. The electrical model is adapted from [23, 24] to include the charger and the motors, as shown in Figure 8.

The electrical model of the motors (Figure 8) is adapted from [25–27]:

$$\begin{aligned} V_{\text{mot}_i} &= I_{\text{mot}_i} \cdot R_m + L_m \cdot \dot{I}_{\text{mot}_i} + K_m \cdot w_{\text{mot}_i}, \quad i \in \{L, R\}, \\ J \cdot \dot{w}_{\text{mot}_i} + K_r \cdot w_{\text{mot}_i} + M_{X_i} &= K_m \cdot I_{\text{mot}_i}, \quad i \in \{L, R\}, \\ M_i &= K_m \cdot I_{\text{mot}_i}, \quad i \in \{L, R\}, \end{aligned} \tag{28}$$

$$V_{\text{mot}_i} = u_i \cdot V_{\text{bat}}, \quad i \in \{L, R\}, \tag{29}$$

$$\text{abs}\left(V_{\text{mot}_i} \cdot I_{\text{mot}_i}\right) = \eta \cdot V_{\text{bat}} \cdot I_{M_i}, \quad i \in \{L, R\}. \tag{30}$$

Regarding motor i, V_{mot_i} (V) and I_{mot_i} (A) are the voltage and the current; R_m is the motor winding resistor (Ω); L_m is the motor inductance (H); K_m is the electromotive constant ($\text{kgm}^2\text{s}^{-2}\text{A}^{-1}$); w_{mot_i} is the rotor angular velocity (rad/s); J is the moment of inertia of the rotor (kgm^2); K_r is the coefficient of rotation resistance; M_{X_i} is the load moment (kg m^2/s^2); η is the performance coefficient; I_{M_i} is the input current (A); and abs is the absolute value function; the control signals of the motors u_L and u_R are calculated by equation (33), and they are the inputs of the motor drivers.

The main equations of the equivalent electrical circuit of the AGV and the charger are as follows:

$$\begin{aligned} I_{\text{bat}} + I_c \cdot u_c &= \frac{V_{\text{bat}}}{R_c} \cdot u_c + I_{\text{stop}} + I_{M_L} + I_{M_R}, \\ I_{R2} &= I_p + I_{\text{bat}}, \\ I_{R2} &= I_{R1} + I_{c1},\ I_{C1} = -C_1 \cdot \dot{V}_C, \\ Em_{\text{bat}} &= V_c + R_2 \cdot I_{R2} + R_0 \cdot I_{\text{bat}} + V_{\text{bat}}, \\ \dot{Em}_{\text{bat}} &= -\frac{I_{\text{bat}}}{(3600 \cdot C_n)}, \end{aligned} \tag{31}$$

where I_{bat} (A) and V_{bat} (V) are the current and the voltage in the terminals of the battery; I_{stop} (A) is the current consumed by the AGV when it is stopped; I_c (A) is the current given by the charging station; R_c (Ω) is the equivalent resistance of the charger;I_p (A) is the parasitic current in the battery when is charging; Em_{bat} (V) is the voltage in the open circuit of the battery; R_1(Ω), R_2(Ω), and C_1(F) are internal parameters of the battery; and C_n (Ah) is the capacity of the battery. The control signal u_c is active, $u_c = 1$, when the AGV is in a charging station. For more details of these variables and parameters, see [23].

4.6. Workspace. The workspace, although is not embedded in the AGV as the rest of the subsystems, plays an important role in the performance of the automatic vehicle. It represents the environment where the AGV moves. It collects information about the ground, the obstacles, and other possible AGVs and provides information to the safety system, the guiding and localization system, the driving system, and the control system.

Figure 9 shows the interaction among these subsystems. The workspace gives the landmarks and references to the GLS to calculate the guiding error err_{gui} and localization. It also detects obstacles that will be handled by the safety system. In addition, it interacts with the driving systems by the friction coefficients and the ramps of the ground. In turn, the driving system moves the AGV, changing its position coordinates (x, y). Finally, the workspace sends IO signals to the control system that may come from other machines [28], AGVs, and human operators, and it also receives commands from the control system.

The workspace can be simulated as a lookup table, where the inputs of the table are the AGV position coordinates (x, y), and the output is a set that provides information about the obstacles, the landmarks and references, the friction coefficients, the ramps, and the IO signals in that spatial location.

5. Simulation Results and Discussion

Several simulations have been carried out to validate the analytical modelling approach [29] and analyze the AGV model. The model has been implemented with Matlab/Simulink software package. In order to reduce the discretization error, a variable simulation step size has been used, with maximum step size set to 25 ms. The control sample time Tc has been also fixed to 25 ms. The AGV parameters used in the simulations are listed in Table 1. The load mass m_L is set to 500 kg.

The simulation scenario is represented in Figure 10. It is a magnetic tape loop (green line) with a charging station and a traffic light. The charging station supplies the energy to the vehicle, so the AGV does not need to leave the circuit to reload. This component, part of the workspace, has been included in the electrical equations of Section 4.5 due to its electrical nature.

A typical element that interacts with the AGV through the IO System is a traffic controller (shown in Figure 10 as a traffic light). When the AGV arrives at an intersection, it stops; then, it requests crossing priority to the traffic

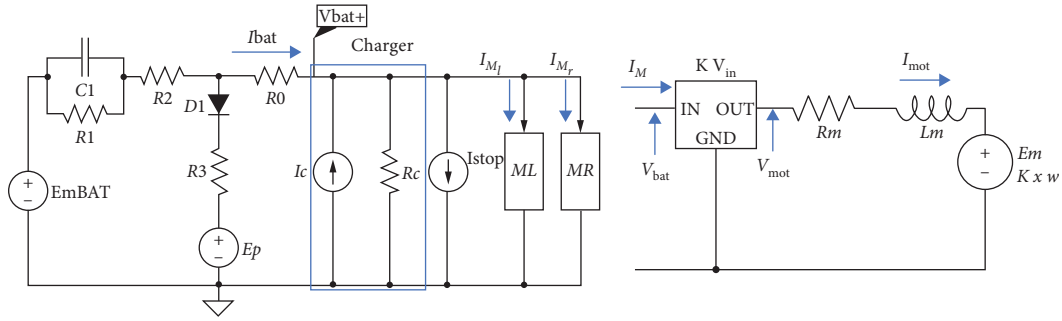


Figure 8: Equivalent circuit of an AGV and charger (a) and of one motor (b).

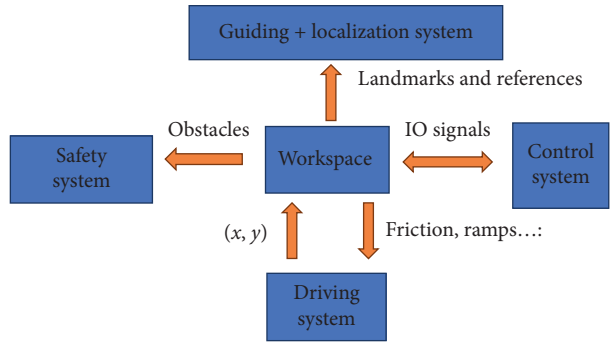


Figure 9: Interaction between the workspace and the AGV subsystems.

controller, and when the intersection is free, the traffic controller informs the AGV and the vehicle continues moving.

The trajectory proposed in Figure 10 is Bernoulli's lemniscate. This trajectory is interesting because the curvature radius is variable and even more, it has an intersection point as it happens in many real layouts. It can be described as the set of P points where the product of the distance to two focus, $F1$ and $F2$, located at a distance $2a$ between them, is a^2.

$$\left(x^2 + y^2\right)^2 - 2a^2\left(x^2 - y^2\right) = 0. \tag{32}$$

The intersection between the straight line projected by the guiding sensor and the lemniscate is used to calculate the guiding error. To calculate the crossing points, we substitute the well-known straight-line equation $y = mx + b$ in (32) and obtain a fourth-degree polynomial (33), with the constants k_0 to k_4 defined in the following:

$$k_4x^4 + k_3x^4 + k_2x^4 + k_1x + k_0 = 0, \tag{33}$$

$$k_4 = m^4 + 2m^2 + 1, \tag{34}$$

$$k_3 = 4m^3b + 4mb, \tag{35}$$

$$k_2 = b^2\left(2 + 6m^2\right) - 2a^2\left(1 + m^2\right), \tag{36}$$

$$k_1 = 4m\left(b^3 + a^2b\right), \tag{37}$$

$$k_0 = b^4 + 2a^2b^2. \tag{38}$$

The solution of this polynomial gives the x coordinate of the intersection points. As it is fourth-degree polynomial, the analytical solution is not evident so it is solved by numerical methods.

During the simulation, it has been supposed that there are not ramps in the environment, all the localizations have the same friction coefficient, and friction coefficients are considered constant due to the large system time constants.

For the sake of clarity, the inputs and outputs of the simulation blocks used during the experiments are shown in Table 2.

5.1. Kinematic Behaviour Simulation. Figure 11 shows the trajectory followed by the traction unit (Figure 11(a)) and the body of the AGV (Figure 11(b)) when reference is a lemniscate curve with a sinusoidal velocity profile, with $v_{\text{ref}} = 0.4$ m/s of amplitude and 0.4 m/s of mean value. In this experiment, the a parameter of the lemniscate is set to 5, and the initial position of the AGV is (3.144, 2.5) with an angle of 0° in the traction unit and the body.

This figure represents 200 steps of the AGV motion on the XY plane, between the initial and the final position. It is possible to observe how the positions described by the traction unit follow a lemniscate-shaped trajectory. This type of representation is very useful to design the layout where the AGV is going to travel in a factory or workhouse. It allows to foresee, among other cases, whether the AGV will be blocked by an obstacle, get stuck in a narrow corridor, or if it will reach correctly a station.

Table 1: Parameters of the wind turbine model.

Parameter	Description	Value/units
L_h	Distance between wheels of the AGV	30 cm
L_b	Distance between rear wheels and traction unit	100 cm
R_h	Radius of front wheels	6 cm
R_b	Radius of rear wheels	9 cm
m_{AGV}	Mass of the AGV	100 kg
I_h	Inertia of traction unit	0,11 kg m^2
C_{roll}	Rolling coefficient	0.01
C_{aero}	Aerodynamic coefficient	0.35
F_{sh}	Static friction coefficient of traction unit	0.1 N
F_{vh}	Viscous friction coefficient of traction unit	0.01 Ns/rad
F_{sw}	Static friction coefficient of traction wheels	$2.94e-2$
F_{vw}	Viscous friction coefficient of traction wheels	$5e-4$ Ns/rad
g	Acceleration of the gravity	9.8 m/s^2
η	Performance coefficient of the motor driver	0.9
I_{stop}	Current when AGV is stopped	1.5 A
R_C	Resistance of the charger	10 KΩ
$[R_0,\ R_1,\ R_2]$	Resistances of the electrical model	[0.05, 0.1, 0.05] Ω
C_1	Capacitance of the battery	1500 F
I_p	Parasitic intensity	50 mA
R_m	Resistance of the motor	1.2 ohm
L_m	Inductance of the motor	$5.8e-4$H
K_m	Electromotive constant of the motor	0.66 $kgm^2s^{-2}A^{-}1$
C_n	Capacity of the battery	150 Ah
$[K_{\text{vp}}, K_{\text{vI}}]$	PID constants for velocity control of wheels	[2, 0.1]
$[K_{\Phi p}, K_{\Phi D}, K_{\Phi I}]$	PID constants for angular velocity control of the traction unit	[9.8, 1, 0.1]

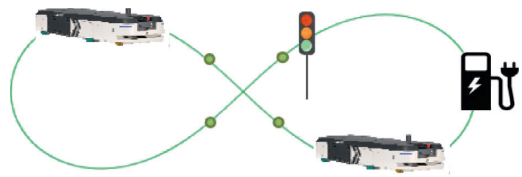


Figure 10: Simulation scenario.

The trajectories followed by the traction unit (red line) and the body (orange line) of the AGV are represented in Figure 12, with the reference (blue line). The trajectory on the plane is shown on the left, and the evolution of each coordinate is represented on the right. The coordinates (x_H, y_H) represent the center of the traction unit, and (x_B, y_B) are the coordinates of the middle of the rear axle.

The traction unit follows reasonably well the lemniscate-shaped magnetic tape; thus, the reference and the traction unit lines are overlapped in Figure 12(a). Nevertheless, the body of the AGV follows a lemniscate curve closer than the reference. This difference decreases with the width of the lemniscate, parameter a in (30). This effect is more visible in the edges of the lemniscate, and it is not shown at the intersection.

Looking closely at Figure 12(b), local maximums and minimums appear at $x \approx 7$ m, $x \approx -7$ m, $y \approx 2.5$ m, and $y \approx -2.5$ m, the extremes of the lemniscate, as expected. It is also interesting to note how when the AGV is in the first and third quadrant, from $t = 0$ to $t \approx 10$ s and from t around 60 s to 100 s, x_H seems to be delayed respect to x_B. However, when it is in the second and fourth quadrants, the opposite happens.

Figure 13(a) represents the same trajectory as in Figure 12 in order to help to understand the angles shown in Figure 13(b). On the right side of this Figure 13, the angle of the traction unit (blue line) and of the body (red line) of the AGV is shown.

The angle of the body of the AGV decreases up to 225° around $t = 28$ s when the straight part of the lemniscate starts, and it keeps constant until that straight part of the lemniscate end about to $t = 40$ s. After this point, the AGV turns counterclockwise, which increases the angle up to approximately $t = 70$ s when the straight part of the lemniscate starts again. On the other hand, the angle of the traction unit is related to the radius of curvature. At the top and bottom edges of the lemniscate, this radius slightly varies and the angle of the traction unit keeps practically constant at −25° from $t = 10$ s up to $t = 25$ s and at 25° from $t = 50$ s up to $t = 65$ s. This angle is positive when the AGV turns clockwise and negative otherwise. In the straight part of the lemniscate, as expected, the angle of the traction unit is 0°.

So, these results validate the model as the AGV moves as expected.

Table 2: Inputs and outputs of the simulation blocks.

Subsystem	Inputs	Outputs
Control system	v_{safe}, err_{gui}, stop, localization	u_l, u_r, $type_{zone}$
Safe system	$type_{zone}$, x_b, y_b, Φ_b, obstacles	v_{safe}, stop
Electrical system	u_l, u_r	M_l, M_r
Driving system	M_l, M_r	$x_b, y_b, \Phi_b, x_h, y_h, \Phi_h, \dot{\theta}_l, \dot{\theta}_r$
Guiding and localization system	x_h, y_h, Φ_h, layout	err_{gui}, localization
Workspace	x_b, y_b	Layout, obstacles

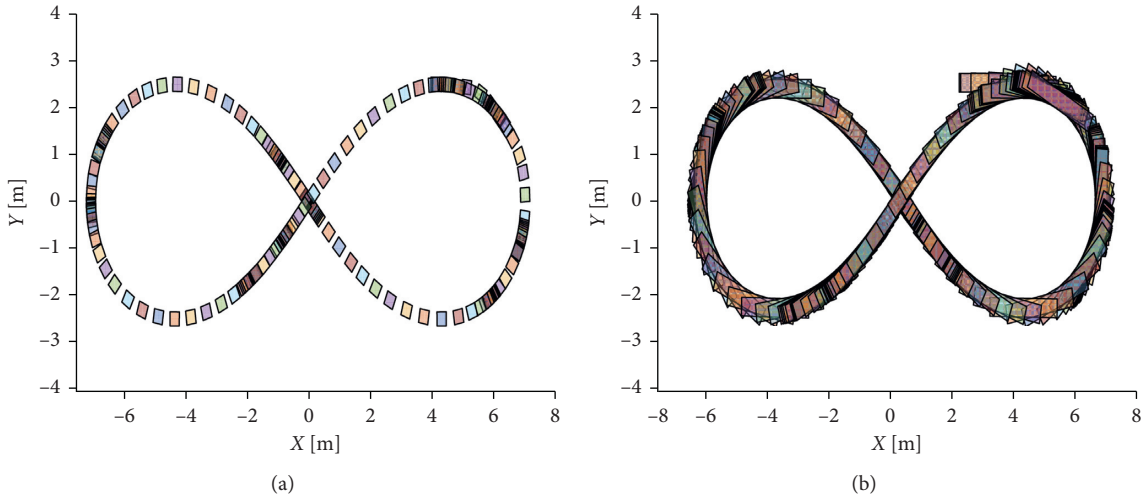


Figure 11: Movement of the traction unit (a) and movement of the body of the AGV (b).

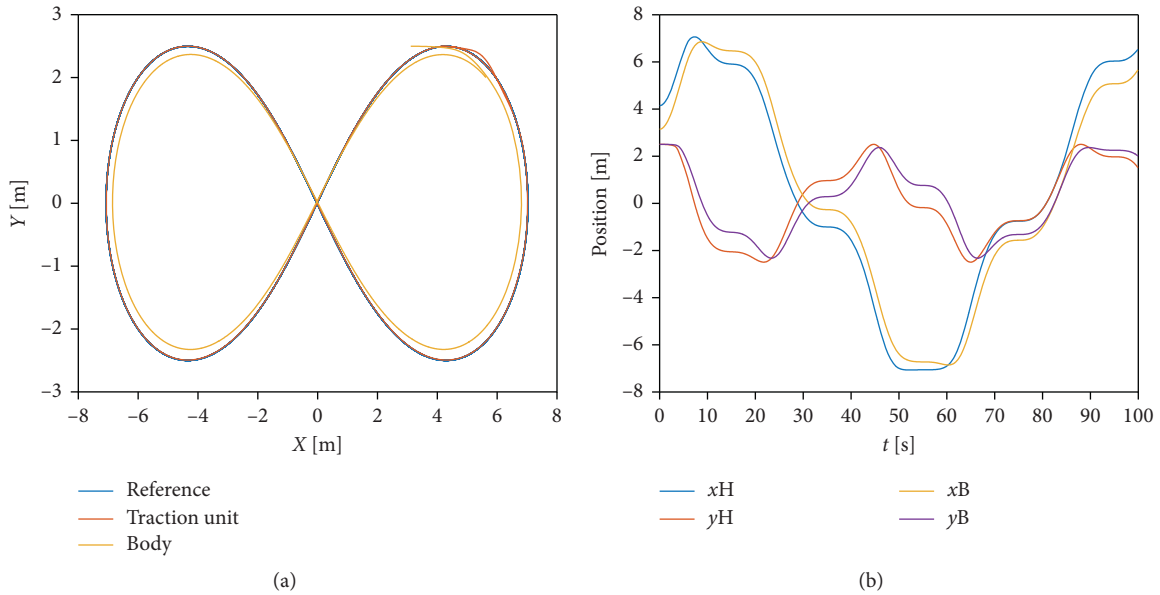


Figure 12: Trajectory of the AGV (a) and *x*-*y* coordinates of the traction unit and the body (b)

5.2. Dynamic Behaviour Simulation. The model of the dynamics of the AGV has been also implemented. Figure 14(a) shows the angular velocity of the traction unit (red line) and its reference (blue line). Figure 14(b) represents the reference of v_{ref} (blue line) and the longitudinal velocity (red line). When the AGV starts moving, the angular speed reference, that is, generated by the controller, oscillates due to error of the velocity. Once the reference speed is reached, at $t = 8$ s, the oscillations are much smaller and the mean value is around 0. In any case, the angular velocity reference is well tracked, with a slight oscillation (smaller than 0.5 rad/s). The sinusoidal profile of v_{foll} (Figure 14(b) is well followed by the

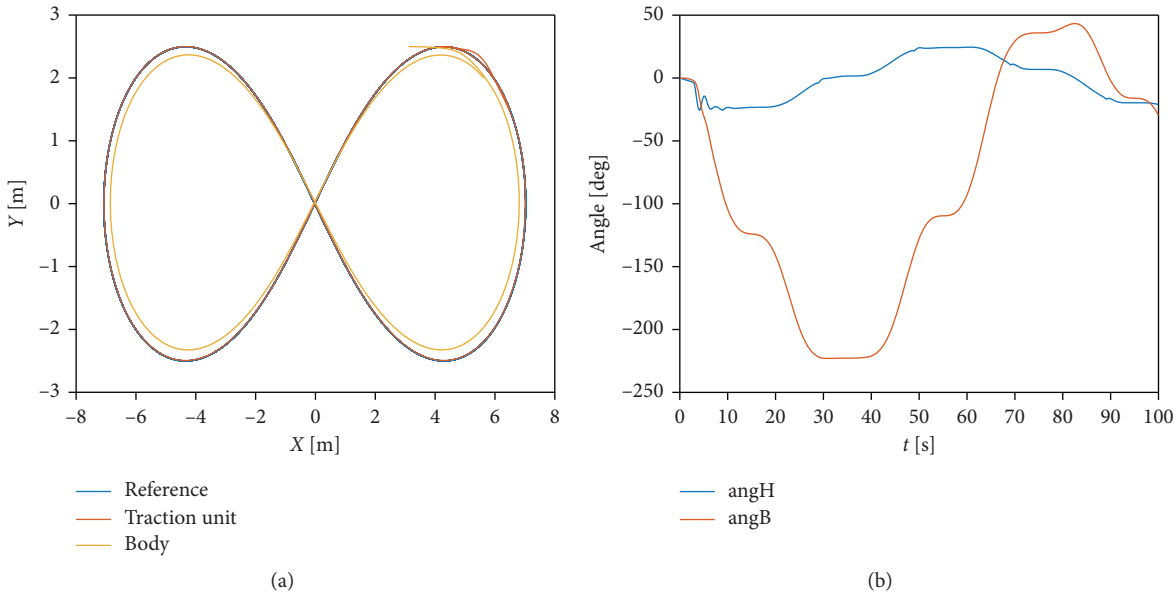


Figure 13: Trajectory of the AGV (a) and angle of the AGV (b).

controller. The biggest error appears at the beginning of the simulation as the AGV is stopped, and it starts accelerating to reach the reference speed.

In order to reach these angular and longitudinal velocities, the corresponding wheel speeds are calculated. Figure 15(a) shows these references, and Figure 15(b) shows the effective wheel velocities. The blue lines represent the right wheel and the red lines the left one. As it is possible to see, the controller is able to track the wheel speed references. The largest error appears at the beginning of the simulation as it is necessary to overcome system inertia. Once the reference value is achieved, the biggest difference between the velocities of the wheels appears at the inflection points of the angle of the traction unit. This happens approximately at $t = 25$ s, 45 s, 65 s, and 85 s. At these points, if the angular velocity of the traction unit is positive, the speed of the right wheel is bigger than the left one, and it is smaller in the opposite case.

In order to follow the wheel speed references (Figure 15(a)), the control has to generate the right power values. Figure 16(a) shows the control signals which are the inputs of the electrical model equation (27); in Figure 16(b), the torque of the wheels is represented. As explained before, this torque is the result of subtracting the friction forces from the torque provided by the motors. The blue lines represent the right wheel and the red lines the left one. The range of the control signals is [−1, 1]. These control signals are inputs in equation (29), and they are dimensionless; they represent the percentage of battery voltage that is applied in the motors.

At the beginning of the simulation, both control signals are saturated to 1, as the initial state of the AGV is stopped. These values are maintained until the wheel speed reference is reached; from this point, the control signals show a sinusoidal behaviour similar to the reference but slightly noisier. Negative peaks appear at t ≈ 30 s, 50 s, 70 s, and 90 s. These values match with the peaks of Figure 14(a). They correct the angular velocity of the traction unit to maintain the angle when the AGV gets close to the top and bottom edges of the lemniscate trajectory. Due to this, the curvature radius is practically constant (Figure 13).

As expected, the shape of the effective torque in the wheels is similar to the control signals but with different amplitude. At the beginning, when the control signals are saturated to 1, the value of the torque is maximum. Then, the effective torque decreases although the control signals are constant due to the effect of the viscous friction forces which increase with the velocity.

5.3. Electrical Behaviour Simulation. In this section, the same experiment is carried out but the analysis is focused on the electrical model. At the initial state, the voltage in the capacitor is $V_c = -4$V, and Em_{bat} is set to 24 V, that are the common values after the charging. The initial current in the motors is 0 A.

Figure 17 shows the current in the motors on the left and the effective torque on the right. The blue lines represent the right motor and the red lines the left one. As expected from equation (26), the absolute value of the effective torque is practically proportional to the current. The difference comes from the friction forces. In the simulations, the current of the motor is considered always positive no matter the sign of the control signal. The sign of the torque in each motor is equal to the sign of its control signal.

In addition to the current of the motors, it is also interesting to study the current and the voltage in the battery terminals. This helps us to scale up the layout and the production cycles, as well as to foresee the battery life. Figure 18 shows the current and voltage in the battery terminals (blue and red lines), the voltage in open circuit of the battery (yellow line), and the voltage in the equivalent capacitor (purple line). As expected, the battery current

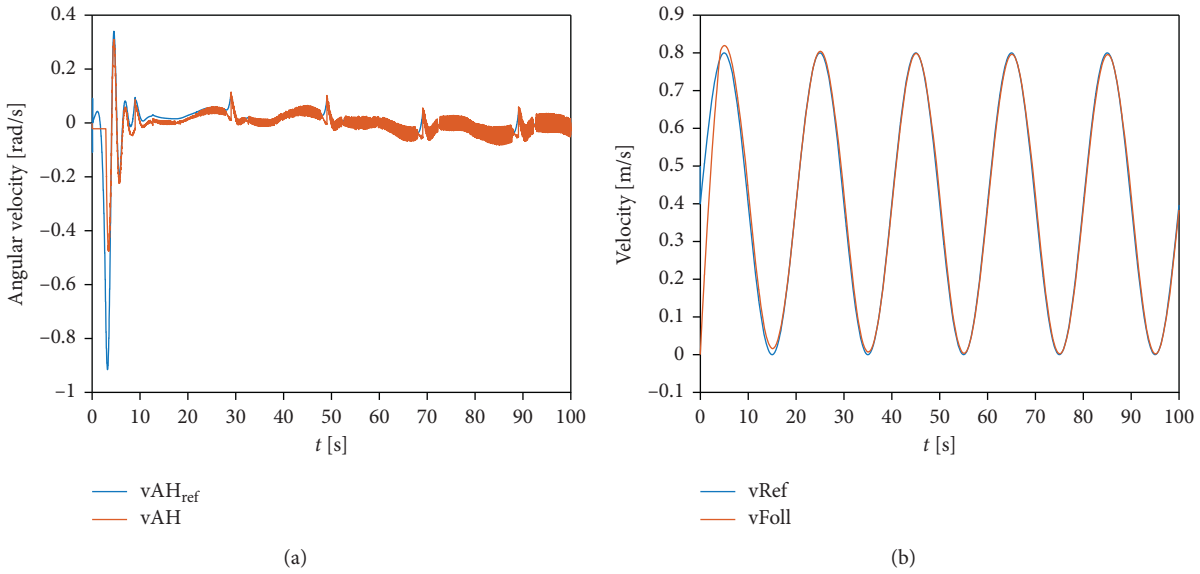


Figure 14: Angular velocity of the traction unit (a) and longitudinal velocity of wheels (b).

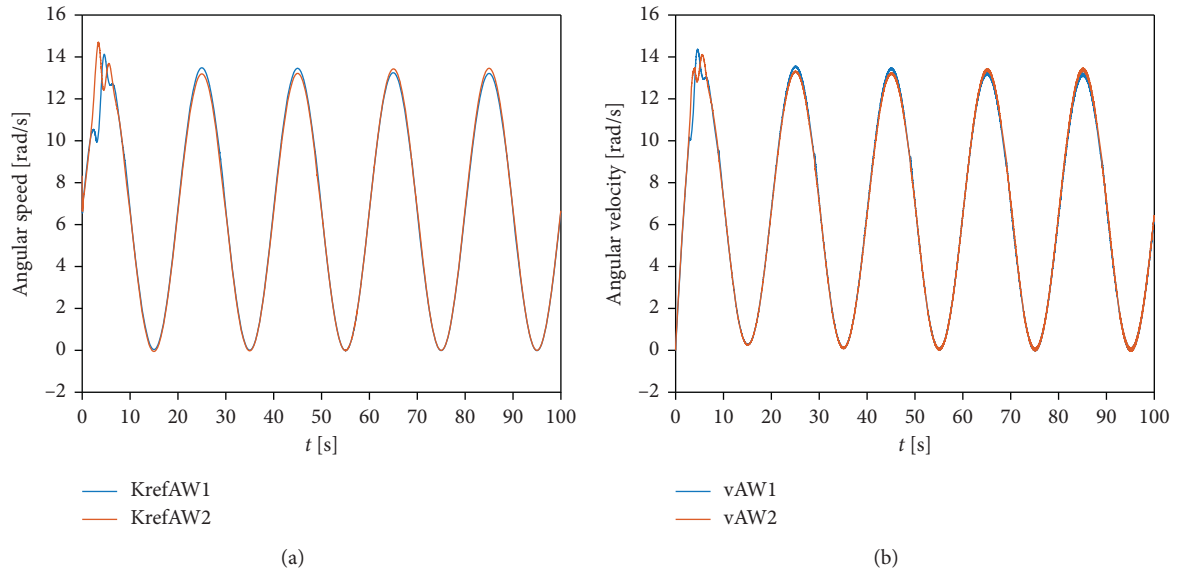


Figure 15: Wheel angular speed references (a) and effective wheel angular velocities (b).

follows a sinusoidal signal. Although sometimes the current in the motors is 0 A, the current in the battery is bigger due to the consumption of the other electronic devices of the AGV. This consumption is considered constant and modelled by I_{stop}. The negative peaks of the torque can be seen in the battery current as positive peaks at the bottom part of the signal.

The battery voltage measured in terminals varies with the current; when the battery current increases, the voltage decreases due to the effect of the internal resistance, producing an observable sinusoidal curve. When the system is working, the absolute value of the voltage in the capacitor decreases. This also produces a decreasing trend in the battery voltage. On the other hand, the Em_{bat} slightly decreases as well but this is not noticeable due to the high capacity of the battery, i.e., the bigger the capacity, the slower the decreasing trend.

Another interesting effect is that when the battery voltage decreases, the voltage applied to the motor for the same control value also decreases. And if the motor voltage decreases, its speed also decreases. This forces to increase the control signal to get the same motor velocity. In turn, the increment of the control makes the consumed current to increase, as it is possible to see in Figure 18. This effect is less significant when the capacitor is discharged since the decreasing trend of the battery voltage is much smaller.

Figure 19 shows the total accumulated current in Ah (Figure 19(a)) and the percentage (%) of the battery (Figure 19(b)). As expected, the sinusoidal current in the battery terminal produces a sinusoidal total current. It is a

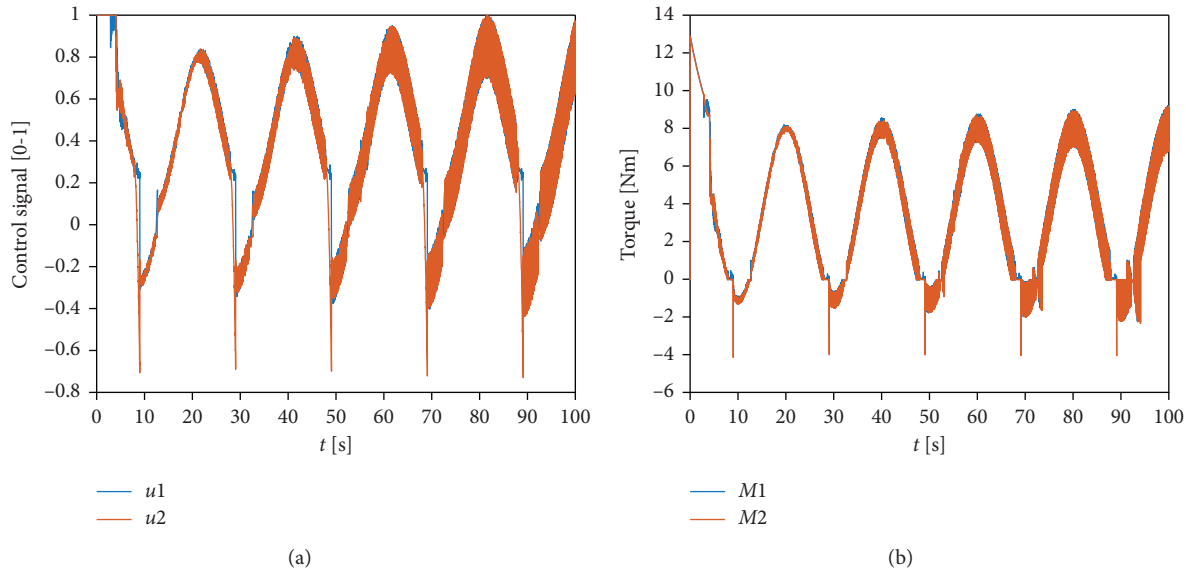


Figure 16: Control signals for each wheel (a) and torque of the wheels (b).

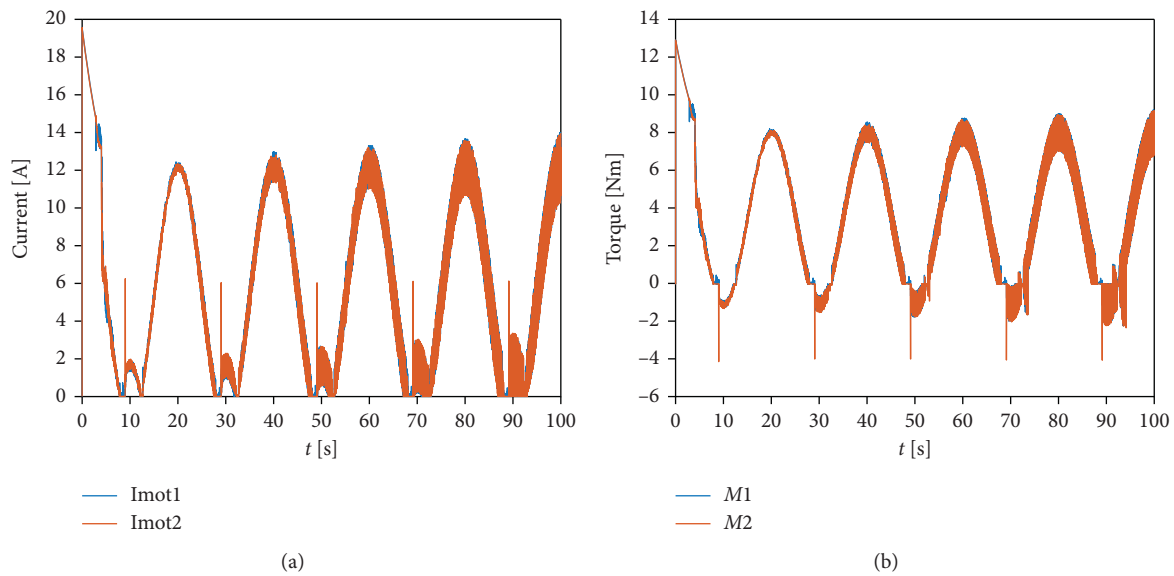


Figure 17: Motor current (a) and effective torque in wheels (b).

usual practice to calculate the percentage of the battery using the total consumed current. In this case, the capacity of the battery is 150 Ah; thus, a total current of 0.3 Ah is equivalent to 0.2% battery, as shown in Figure 19(b). This provides an autonomy of approximately 13.8 h. This is an approximation since it has been seen, the battery voltage also affects the current.

5.4. Influence of the Mass of the Load. The effect of the load on the AGV has also to be analyzed. To do it, different experiments varying the load mass have been carried out with two different speed profiles: a sinusoidal one, as before, and a constant speed of 0.8 m/s. All these experiments are configured as in the previous sections but the mass. In the previous experiments, the load mass m_L was set to 500 kg.

Figure 20 represents the variation of the average battery power with the mass (Figure 20(a)) and the variation of the average current (Figure 20(b)). The blue lines represent the values for a sinusoidal speed profile and the red lines for a constant velocity. As expected, the bigger the load mass, the larger the power and current. This effect is more marked when the speed is constant. As the AGV travels more time at a bigger speed, it needs more energy.

Figure 21 shows the variation of the MSE guiding error (Figure 21(a)) and the variation of the standard deviation of the total torque in the wheels (Figure 21(b)). The color code

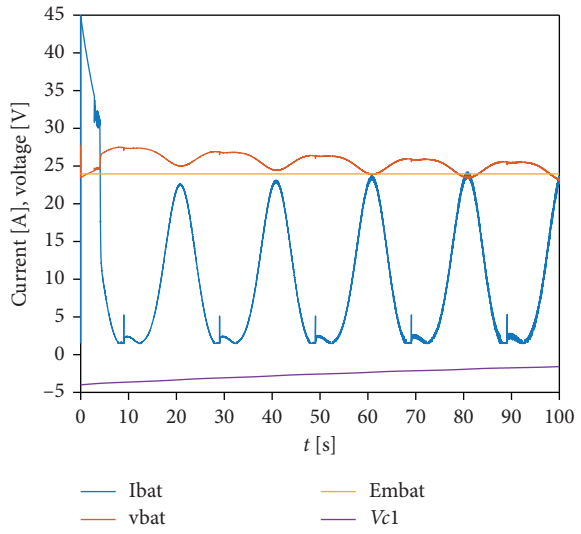


Figure 18: Current and voltage in the battery.

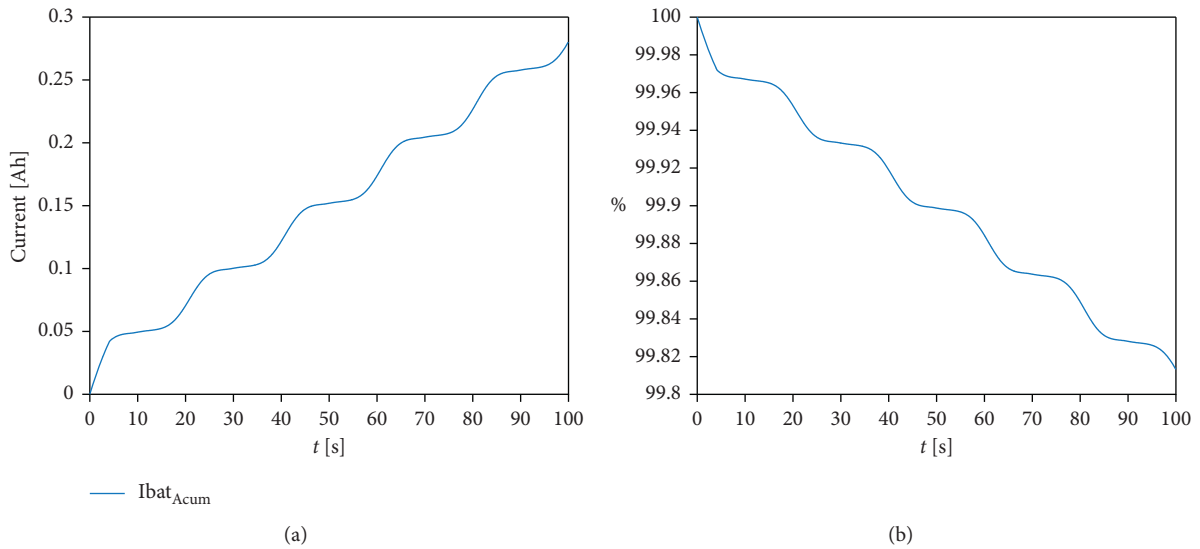


Figure 19: Total consumed current (a) and percentage of battery (b).

is the same as in Figure 20. These indicators have been calculated using the following equations:

$$\mathrm{MSE}\,[\mathrm{cm}] = 100\sqrt{\frac{1}{T_{\mathrm{sim}}}\sum_{i}\left(\mathrm{err}_{\mathrm{gui}_i}\right)^2 T_{s_i}}, \tag{39}$$

$$\overline{M_e}\,[\mathrm{Nm}] = \frac{1}{T_{\mathrm{sim}}}\sum_{i}\left(\left|M_{eL_i}\right| + \left|M_{eR_i}\right|\right)T_{s_i}, \tag{40}$$

$$\mathrm{SD}_{M_e}\,[\mathrm{Nm}] = \sqrt{\frac{1}{T_{\mathrm{sim}}}\sum_{i}\left[\left(\left|M_{eL_i}\right| + \left|M_{eR_i}\right|\right) - \overline{M_e}\right]^2 T_{s_i}}, \tag{41}$$

where T_{s_i} is the sampling time at i and T_{sim} is the total simulation time. This period T_{s_i} is necessary since a variable simulation step has been used. This way the MSE gives the mean squared value of the guiding error in cm. The standard deviation of the total torque can give us an idea of the degradation of the electromechanical pieces, as larger total torque means bigger degradation.

In Figure 21, it is possible to observe how the MSE is not affected by the mass until certain mass threshold is surpassed. This value is around 300 kg for the sinusoidal speed profile and 150 kg for the constant speed. This may be explained as the average velocity with the sinusoidal profile is approximately half than the obtained with the constant speed. However, the effect of the mass on the standard deviation of the torque is worse. The sinusoidal speed profile produces bigger deviations, and moreover, the influence of the mass is bigger.

5.5. Influence of the AGV Velocity. Finally, the effect of using two AGV velocity profiles, sinusoidal and constant, is

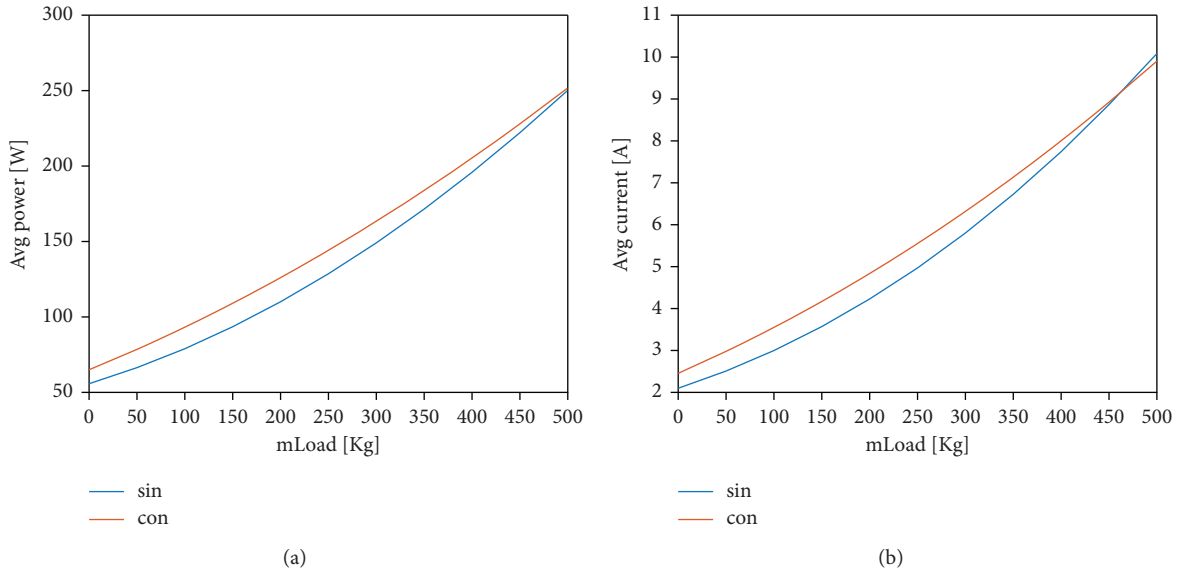


Figure 20: Variation of the average power (a) and average current (b) with load mass.

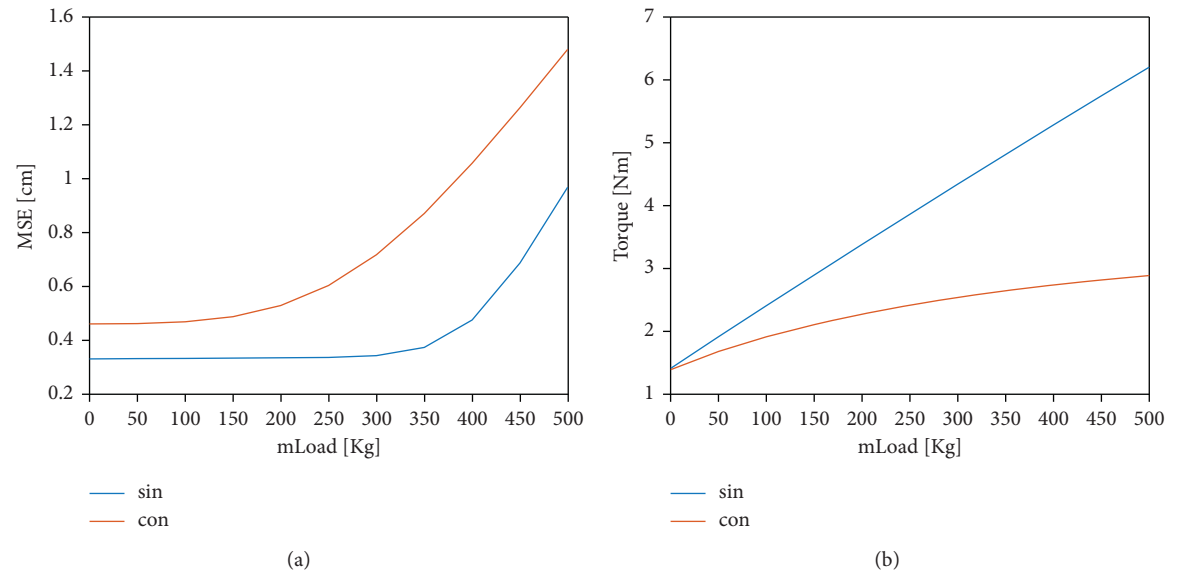


Figure 21: Variation of the MSE (a) and the standard deviation of the torque (b) with the load mass.

studied. The mean amplitude of the sinusoidal signal is half the value fixed for the constant one. The load mass is 500 kg. The rest of parameters of the experiment have been configured as in previous sections.

Figure 22 represents the variation of the average battery power with the velocity (Figure 22(a)) and the variation of the average current (Figure 22(b)). The color code is the same as in the previous experiments. Again, as expected, bigger velocity produces larger power and bigger current. However, in this case, the sensitivity, defined as the derivative of the measured variable respect to the parameter that is modified by the sweep, is smaller.

The sensitivity is denoted by $\beta_{\text{par}}^{\text{var}}$, where var is the measured variable and par is the parameter that is modified in the experiment. In particular, $\beta_{m_L}^{\text{Pow}}$ is 0.4 W/kg, $\beta_{v_{\text{foll}}}^{\text{Pow}}$ is 0.18 W·s/cm, $\beta_{m_L}^{I_{\text{bat}}}$ is around 0.016 W/kg, and $\beta_{v_{\text{foll}}}^{I_{\text{bat}}}$ is 0.007 W·s/cm, approximately.

Figure 23 shows the variation of the MSE (Figure 23(a)) and of the standard deviation of the total torque in the wheels (Figure 23(b)). The color code is the same as in Figure 22. It is remarkable how the error is related to the cruise velocity; lower speeds produce smaller errors. This is so in general, but there exist an inferior limit related to the accuracy of the encoders that measure the velocity. The velocity is also related to the torque, as shown in Figure 23(b), where larger velocities demand bigger effective torque. This is not a general rule since the torque is directly related to the acceleration; in order to increase the

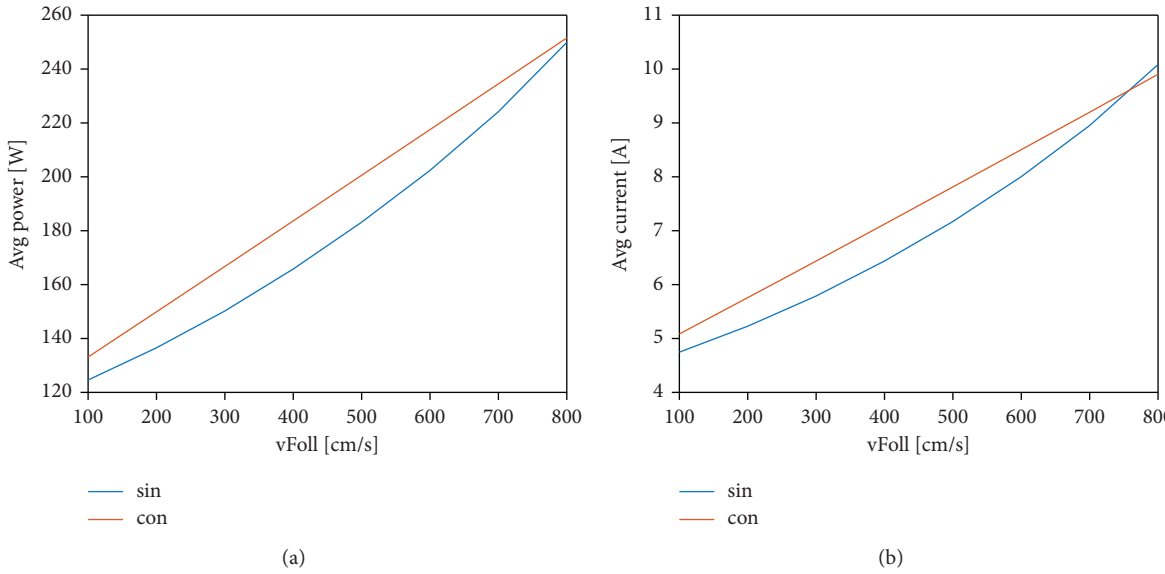


Figure 22: Variation of the average power (a) and average current (b) with the cruise velocity.

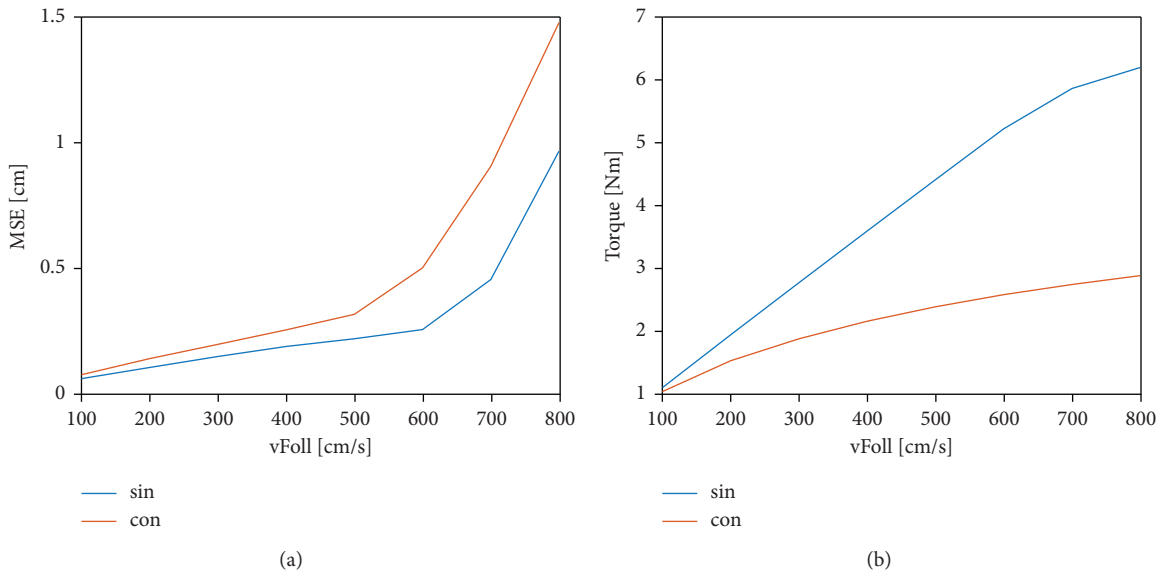


Figure 23: Variation of the MSE (a) and the standard deviation of the torque (b) with the cruise velocity.

acceleration of the AGV, it is necessary to increase the effective torque of the wheels. However, a bigger velocity does not necessarily entail a bigger torque since a higher velocity can be reached with the same acceleration and torque but spending more time. However, in this case, the AGV cannot leave the magnetic circuit; thus, bigger accelerations are necessary to maintain the cruise velocity and that increases the effective torque.

6. Conclusions and Future Works

Automatic guided vehicles are complex systems with highly coupled internal variables. Their use has lately grown exponentially due to the Industry 4.0 paradigm, as these are industrial devices and play a key role in process flow and in the logistic sector. However, some of their safety requirements depend on the dynamics and control of these vehicles. Therefore, a holistic modelling approach of the AGV is necessary in order to better understand its performance and to design efficient controllers.

In this paper, a complete model of a hybrid AGV has been developed. The vehicle is a general one, widely used, that is a mixture of tricycle and differential robots. The different subsystems of the AGV have been described and mathematically modelled, particularly, the control system, the safety system, the driving system, the guiding and location system, the charging, the power storage, and the motors. Even more, the interactions among them have been identified and represented in the proposed architecture of the AGV.

Simulations results, varying some of the vehicle parameters, have allowed it to validate the models as the global performance of the AGV is as expected.

Among other possible future works, we may highlight the study of the influence of the deformation of the wheels and the design of neural-controllers to improve the performance provided by the current PIDs. Besides, rollover conditions depending on the load could be estimated to ensure the safety of these systems.

Data Availability

The equations and parameters used in this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was partially supported by the European Commission, under European Project CoLLaboratE, grant no. 820767 [2].

References

[1] J. E. S. Garcia, F. Espinosa, and C. Santos, "Multi-AGV transport of a load: state of art and centralized proposal," *Revista Iberoamericana de Automática e Informática industrial*, vol. 2020, 2020.

[2] CoLLaboratE, 2020, https://collaborate-project.eu/.

[3] J. Gonzalez, X. Álvarez, A. M. González et al., "Fleet planning of AGVs," 2016.

[4] ASTI Mobile Robotics, 2019, https://asti.es/es/agvs#easybots.

[5] W. L. Maxwell and J. A. Muckstadt, "Design of automatic guided vehicle systems," *IIE Transactions*, vol. 14, no. 2, pp. 114–124, 1982.

[6] J. Veiga, J. Sousa, J. Machado, J. Mendonça, T. Machado, and P. Silva, "Modeling of dynamic behavior of AGV systems," 2019.

[7] M. Bonilla, F. Reyes, and M. Mendoza, "Modelling and simulation of a wheeled mobile robot in configuration classical tricycle," 2005.

[8] J. Villagra and D. Herrero-Pérez, "A comparison of control techniques for robust docking maneuvers of an AGV," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 4, pp. 1116–1123, 2011.

[9] Y. Han, Y. Cheng, and G. Xu, "Trajectory tracking control of AGV based on sliding mode control with the improved reaching law," *IEEE Access*, vol. 7, pp. 20748–20755, 2019.

[10] D. A. Aligia, G. A. Magallán, and C. H. De Angelo, "Control de Tracción para un Vehículo Eléctrico basado en Observadores no Lineales," *Revista Iberoamericana de Automática e Informática industrial*, vol. 15, no. 1, pp. 112–123, 2017.

[11] X. Wu, C. Sun, T. Zou, H. Xiao, L. Zhai, and J. Zhai, "Intelligent path recognition against image noises for vision guidance of automated guided vehicles in a complex workspace," *Applied Sciences*, vol. 9, no. 19, p. 4108, 2019.

[12] M. Smieszek, M. Dobrzanska, and P. Dobrzanski, "Measurement of wheel radius in an automated guided vehicle," *Applied Sciences*, vol. 10, no. 16, p. 5490, 2020.

[13] R. Stetter, M. Witczak, and M. Pazera, "Virtual diagnostic sensors design for an automated guided vehicle," *Applied Sciences*, vol. 8, no. 5, p. 702, 2018.

[14] T. Kawakami and S. Takata, "Battery life cycle management for automatic guided vehicle systems," *Design for Innovative Value towards a Sustainable Society*, vol. 8, pp. 403–408, 2012.

[15] L. Hou, L. Zhang, and J. Kim, "Energy modeling and power measurement for mobile robots," *Energies*, vol. 12, no. 1, p. 27, 2019.

[16] I. F. A. Vis, "Survey of research in the design and control of automated guided vehicle systems," *European Journal of Operational Research*, vol. 170, no. 3, pp. 677–709, 2006.

[17] A. Bayuwindra, J. Ploeg, E. Lefeber, and H. Nijmeijer, "Combined longitudinal and lateral control of car-like vehicle platooning with extended look-ahead," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 3, pp. 790–803, 2019.

[18] S. Baldi, D. Liu, V. Jain, and W. Yu, "Establishing platoons of bidirectional cooperative vehicles with engine limits and uncertain dynamics," *IEEE Transactions on Intelligent Transportation Systems*, vol. 2, 2020.

[19] P. Ghafoorpoor Yazdi, A. Azizi, and M. Hashemipour, "A hybrid methodology for validation of optimization solutions effects on manufacturing sustainability with time study and simulation approach for SMEs," *Sustainability*, vol. 11, no. 5, p. 1454, 2019.

[20] Oriolo, G. Control of Nonholonomic Systems, 2020, https://www.dis.uniroma1.it/~oriolo/cns/cns_slides.pdf.

[21] T. A. Henzinger, "The theory of hybrid automata," *Verification of Digital and Hybrid Systems*, vol. 23, pp. 265–292, 2000.

[22] S. Ito, M. Soga, S. Hiratsuka, H. Matsubara, and M. Ogawa, "Quality index of supervised data for convolutional neural network-based localization," *Applied Sciences*, vol. 9, no. 10, p. 1983, 2019.

[23] R. A. Jackey, "A simple, effective lead-acid battery modeling process for electrical system component selection," *SAE Transactions*, vol. 9, pp. 219–227, 2007.

[24] F. J. Gómez, L. J. Yebra, A. Giménez, and J. L. Torres-Moreno, "Modelado de baterías para aplicación en vehículos urbanos eléctricos ligeros," *Revista Iberoamericana de Automática e Informática industrial*, vol. 16, no. 4, pp. 459–466, 2019.

[25] F. Dušek, D. Honc, and P. Rozsíval, "Mathematical model of differentially steered mobile robot," 2011.

[26] E. Amaya-Cruz, O. Gutierrez-Frias, A. Luviano-Juarez, and H. Sossa-Azuela, "Design and construction of a robotic platform for 3D reconstruction through an embedded processing system," *IEEE Latin America Transactions*, vol. 16, no. 1, pp. 19–24, 2018.

[27] A. Beltrán, J. Rumbo, H. Azcaray, K. Santiago, M. Calixto, and E. Sarmiento, "Simulation and control of the velocity and electromagnetic torque of a three-phase induction motor: an electric vehicles approach," *Revista Iberoamericana de Automática e Informática industrial*, vol. 16, no. 3, pp. 308–320, 2019.

[28] F. Espinosa Zapata, J. L. Lázaro Galilea, and J. Olivares Bueno, "Proyecto ALCOR: contribuciones a la Optimización del Guiado Remoto de Robots en Espacios Inteligentes," *Revista Iberoamericana de Automática e Informática industrial*, vol. 15, no. 4, pp. 416–426, 2018.

[29] J. E. Sierra and M. Santos, "Modelling engineering systems using analytical and neural techniques: Hybridization," *Neurocomputing*, vol. 271, pp. 70–83, 2018.