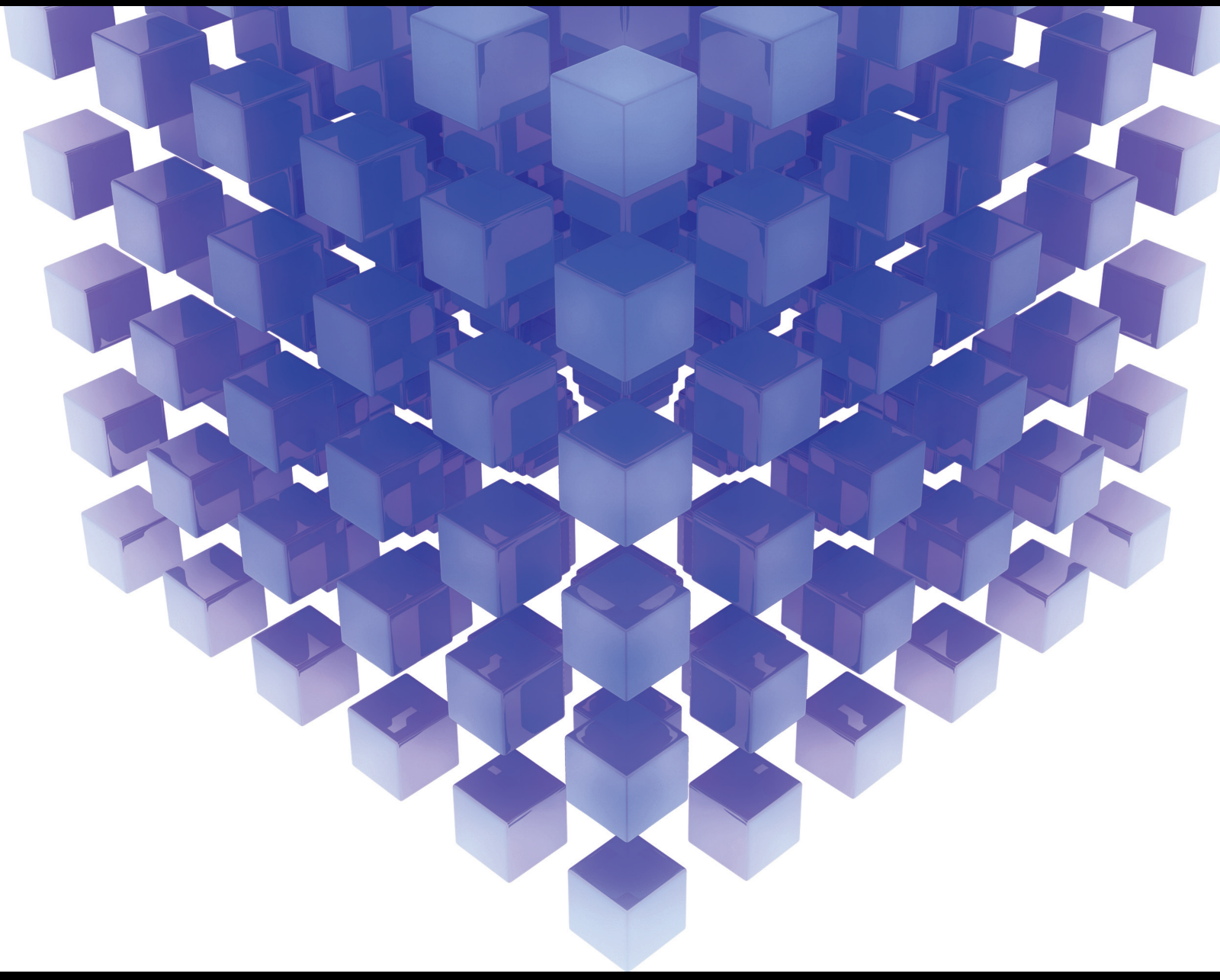
Mathematical Problems in Engineering

Recent Advances in Optimization Theory, Methods, and Applications in Science and Engineering 2021

Lead Guest Editor: Guoqiang Wang
Guest Editors: Jiyuan Tao, Goran Lesaja, and Mohamed El Ghami



Recent Advances in Optimization Theory, Methods, and Applications in Science and Engineering 2021

Mathematical Problems in Engineering

Recent Advances in Optimization Theory, Methods, and Applications in Science and Engineering 2021

Lead Guest Editor: Guoqiang Wang
Guest Editors: Jiyuan Tao, Goran Lesaja, and Mohamed El Ghami

Chief Editor

Academic Editors

Contents

Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 2502346, 16 pages
https://doi.org/10.1155/2022/2502346

Research Article

Nonlinear Stochastic Multiobjective Optimization Problem in Multivariate Stratified Sampling Design

Shokrya Saleh A. Alshqaq,[1] Abdullah Ali H. Ahmadini,[1] and Irfan Ali[2]

[1]*Department of Mathematics, College of Science, Jazan University, Jazan 45142, P.O. Box 114, Saudi Arabia*
[2]*Department of Statistics & Operations Research, Aligarh Muslim University, Aligarh, India*

Correspondence should be addressed to Irfan Ali; irfii.st@amu.ac.in

Received 24 December 2021; Accepted 28 June 2022; Published 29 August 2022

Academic Editor: Jiyuan Tao

Decision-making in survey sampling planning is a tricky situation; sometimes it involves multiple objectives, with various decision variables emanating from heterogeneous and homogeneous populations. Dealing with the entire population under study and its uncertain nature becomes a challenging issue for researchers and policymakers. Hence, an appropriate sampling design and optimization methodology are imperative. The study presents a useful discussion on stochastic multiobjective multivariate stratified sampling (MSS) models theoretically, and the concepts are illustrated with numerical examples. Also, it has been found that the linearization of sampling variance in survey sampling does not help determine the optimal sampling allocation problem with minimum variability. Optimal allocation problems under the weighted goal programming, stochastic goal programming, and Chebyshev goal programming methods are also discussed with numerical examples. Finally, the study discussed the linear approximation of the MSS problem with examples. The study is a conceptual and theoretical framework for MSS under a stochastic environment. The numerical data is simulated using the stratifyR package.

1. Introduction

The classical method of optimization for differential calculus is too restrictive and challenging in terms of applicability to many statistical areas in a real-life situation. The lack of numerical algorithms suitable for solving optimization problems poses some severe limitations in this regard and hence led to the utilization of some inefficient statistical procedures in choosing the objective functions and constraints. For decades, a better technique for optimization with broader applicability in statistics, with an increasing computing power able to be implemented has been forthcoming. Mathematical programming is one of such evolving methods with potential application in statistical methodologies. Several optimization techniques have various applications in statistical problems such as designing a specific experiment, extensive survey for data collection, characterizing observed data using a model, drawing inferences about a population based on sample data, testing of hypothesis, and estimation in the decision-making process [1]. In all the applications, one has to optimize (minimize or maximize) an objective function subject to a set of constraints, such as cost or other input parameters. The sampling problem about the population characteristics remains deriving information on several populations statistically. In a sampling survey, the objectives are to minimize the sampling variance and cost, these depend on the sample size, the sampling scheme, the size of the sampling unit or the scope of the study. Alternatively, a different formulation may be to minimize survey inaccuracy, given that the survey cost is within the budgetary limits. Thus, the research aims at finding a solution for this challenging problem of optimal sample size or sampling scheme that could help in estimating the desired characteristics of a population under prescribed properties. The objective of this research is to successfully formulate the problems of sample surveys as

mathematical programming problems and develop an efficient algorithm or technique to solve them. The objective is to identify existing and future works of allocation problems in survey sampling and to investigate and suggest solutions to them, and also, study the problems in an uncertain environment, i.e., a stochastic. The uncertainty that exists in real life has motivated us to work on this aspect. The problems become more complicated when some or all of the parameters involved are uncertain; it may be either stochastic or fuzzy. The objective is to develop an efficient algorithm to solve such types of sample surveys. The formulated problems may be single-objective or multiobjective. For solving the multiobjective optimization problems, we need to develop efficient algorithms for the formulated problems. The goal programming, fuzzy goal programming, and other new modified version or extended version of these techniques will be used to solve the multiobjective optimization problems.

This study comprises modeling and optimization of different sampling design that helps in providing the efficient allocation of samples simultaneously by achieving the highest accuracy and minimizing the sampling variances. This project provides a useful insight into the decision-making for implementing strategies in different socio-economic sectors for the country based on sampling results. Our contribution to this project is proposing new models and techniques for the sampling scheme to determine the optimal allocation of samples based on which policymakers can suggest what kinds of additional efforts can simultaneously be taken in the planning of socio-economic sectors. The problems related to the case studies are usually complicated, but it has become more complicated when some or all of the input information parameters involved are uncertain. The study provides mathematical optimization problems in survey sampling, which is a powerful tool to make the best policy on national planning and industries. Therefore, the study is an integration of sample surveys, operational research, and computational modeling. Optimal sampling techniques can play an essential role in annual budgeting, income and expenditure forecasting for the preparation of five-year plans in national planning and budgeting. It can also be used in major projects scheduling of national interest, estimation of country's population, agricultural yields, employment, gross national product (GNP), and gross domestic products (GDP) amongst others. Optimal sampling techniques provide the best (optimal) solutions to the problem under study.

There is a need for Statistical Information in modern society now more than any time before, in particular, when data is to be collected periodically to satisfy the information need on a specified set of elements, known as finite populations. Surveys played a significant role in issues relating to real life, if we want to get a sense of a massive population. Sampling is the best tool that gives us a fresh idea about the whole population. A sample survey is one of the most critical data collection modes for meeting this need. Over time, an extensive literature survey sampling has developed into a vast array of theories, processes, and operations that are used every day throughout the globe. It is appropriate to speak of a worldwide survey industry with different sectors, namely, a government sector, academic sector, a private and mass media sector, a residual sector consisting of ad-hoc and in-house surveys. Optimization is the science of selecting the best among many possible decision alternatives in a complicated real-life situation. The ultimate target of any decision is to either maximize the desired benefit or to minimize the effort (cost or time) required or incurred in a particular course of action. In recent times, more authors formulated different types of sampling problem as a nonlinear mathematical programming problem or integer programming problem and tried to find the best solution [2]. An integer compromise allocation in MSS has been determined using the goal programming approach [3]. A multiobjective all-integer nonlinear model for MSS design considering some travelling costs has been developed, and a compromising solution was obtained using the value function approach, ε – constraint method, and distance-based method [4]. Also, uncertainties in the MSS problem have been investigated where some cost parameters were considered as fuzzy parabolic numbers [5]. The authors formulated a fuzzy multiobjective nonlinear programming problem with a quadratic cost function and solved using fuzzy programming. A case of nonresponse in the MSS problem has been studied and modeled as an all-integer multiple objective problem [6]. The solution was sought using four different procedures. Several authors have worked in optimum allocation problems of sampling and parameter estimation, for instance a Multiobjective Integer Nonlinear Programming Problem has been formulated and converted to a single-objective problem using the value function technique [7]. Also, they used Lagrange Multipliers Technique to obtain the continuous sample sizes formula, which approximates the optimal solutions. Similarly, traveling costs within strata has been considered, and a multiple objective nonlinear stochastic programming problem was formulated for finding a compromise allocation in the sample survey [8]. The problem was solved using D_1 – distance, goal programming, and the Chebyshev approximation technique. A problem of estimating p – population means considering nonresponse and nonlinear cost functions have been investigated, and solution procedures suggested using lexicographic goal programming [9]. The dynamic programming technique has been employed in proposing an efficient methodology for optimum stratum boundaries and determining optimum sample size in survey variables under the Neyman allocation [10]. A multiple pooling of the standard deviations of the estimates in an MSS for more than three strata has been studied and formulated as a multiobjective (MO) problem, which was solved using fuzzy programming [11]. Others considered compromise allocation problems under stratified samples with two-stage randomized and multiresponse models [12, 13]. An optimum allocation problem in MSS has been considered as an integer nonlinear stochastic programming and

solved with five different techniques [14]. The authors suggested the use of coefficient of variations instead of variances. Also, the MSS problem has been studied with stochastic optimal design [15, 16], with flexible goals [17], and with integer solution [18].

Several mathematical models have been designed based on multiobjective optimization for solving different aspects of human endeavors. For instance, a mixed-integer linear programming (MILP) model has been developed for addressing a closed-looped supply chain network problem during the coronavirus pandemic. The study considered different items such as recycling, reusing, quarantine, collection, distribution, production, supply, and location within a multiperiod, multiechelon, and multiproduct supply chain [19]. A multiple criteria decision-making tool has been used in determining the supply chain performance in a petrochemical industry incorporating sustainable strategies [20]. An optimization method has been designed to optimize the distribution and allocation of scarce resources amongst individuals during a crisis, based on credibility theory and a harmony search algorithm considering random simulation [21]. A scheduling problem has been studied, and a mathematical model developed with a view to obtain near-optimal solution using meta-heuristic algorithms (MHA) [22]. Multiobjective optimization has been widely used in different sectors considering diverse applications and scenarios. For instance, robust optimization with artificial intelligence (AI) has been hybridized as multiobjective optimization applied to the product portfolio problem [23]. Location, allocation, and routing problem have been studied with the help of an improved harmony search algorithm [24]. Another important application area is that of dairy product's demand prediction, where an integrated approach based on AI and novel MHA has been used in achieving the desired future demands [25]. The MOOP has been used to formulate socio-economic and environmental issues related to sustainable development goals in several countries, such as India [26], Nigeria [27], Saudi Arabia [28], and other areas, such as municipal waste management system [29].

1.1. Organization of the Paper. The introduction of the subject matter, the background of the study, the literature review, and paper organization are presented in section 1. In Section 2, the multiobjective MSS techniques are presented. Section 3 provides single-objective stochastic MSS models. Section 4 discussed the MO stochastic MSS models. The linear approximation of MSS is discussed in Section 5. Section 7 concludes the article.

2. Multiobjective Multivariate Stratified Sampling

Let N be the size of the population partitioned into L strata each of sizes N_h, $h = 1, 2, \ldots, L$. Suppose p is considered as characteristics ($p \geq 2$) that are measured on each unit of the population, and the interest is on p-population characteristics estimation. Let n_h, $h = 1, 2, \ldots, L$ be the units taken randomly from the stratum without replacement.

2.1. Sampling Variance Function. The population mean ($\overline{Y}_j$) for the j^{th} character is

$$\begin{aligned}\overline{Y}_j &= \frac{1}{N}\sum_{h=1}^{L}\sum_{i=1}^{N_h} y_{jhi} \\ &= \sum_{h=1}^{L} W_h \overline{y}_{jh}; \; j = 1, 2, \ldots, p,\end{aligned} \tag{1}$$

where $W_h = N_h/N$ is stratum weights and $\overline{y}_{jh} = (1/N_h)\sum_{i=1}^{N_h} y_{jhi}$ is a stratum means.

The sampling means of the j^{th} character is given as

$$\overline{y}_{j,st} = \frac{1}{n}\sum_{h=1}^{L}\sum_{i=1}^{n_h} y_{jhi} =; \; j = 1, 2, \ldots, p. \tag{2}$$

The sampling variance of the estimator of the mean for the j^{th} characteristic is given as follows:

$$\overline{y}_{j,st} = \sum_{h=1}^{L}\left(\frac{1}{n} - \frac{1}{N_h}\right) W_h^2 S_{jh}^2; \; j = 1, 2, \ldots, p, \tag{3}$$

where $S_{jh}^2 = (1/N_h - 1)\sum_{i=1}^{N_h}(y_{jhi} - \overline{Y}_{jh})^2$ are stratum variances and y_{jhi} is the value of the i^{th} unit in the h^{th} stratum for the j^{th} characteristics ($j = 1, 2, \ldots, p$ and $h = 1, 2, \ldots, L$).

2.2. Sampling Cost Function. In survey sampling, when enumeration cost, traveling cost, and labor costs are high [30, 31], the total cost function is defined as follows:

$$C = \sum_{h=1}^{L} c_h n_h + \sum_{h=1}^{L} t_h \sqrt{n_h} + \omega \sum_{h=1}^{L} \frac{n_h}{\lambda}, \tag{4}$$

where c_h is the per-unit cost of measurement in the h^{th} stratum, t_h is the travel cost for enumerating for a unit of the j-th character in the h-th stratum, and ω is the cost of labor for a unit time. The labor time is available with respect to the time for a sampling unit within a stratum and follows an exponential distribution with rate λ.

If in (4) the labor expenses are not significant, then we have a quadratic cost function given in (5).

$$C = \sum_{h=1}^{L} c_h n_h + \sum_{h=1}^{L} t_h \sqrt{n_h}. \tag{5}$$

If in (5) the traveling cost is not significant, then we have a linear cost function with a fixed overhead cost of sampling (C_0) given in (6).

$$C = \sum_{h=1}^{L} c_h n_h + C_0. \tag{6}$$

In a particularly important case of (6), if $c_h = c$, that is, if the per-unit cost in all strata is assumed to be the same, then the enumeration cost terms become constant, and the fixed cost for optimum allocation reduces to fixed sample size optimum allocation and is called Neyman [32].

2.3. Multiobjective Optimization Problem. The multiobjective optimization problem (MOOP) using the above-given definitions can be defined as

$$\left.\begin{array}{ll} \min f_j(n_h) = \sum_{h=1}^{L}\left(\frac{1}{n_h} - \frac{1}{N_h}\right)W_h^2 S_{jh}^2 & \text{(i)} \\ \text{subject to :} & \\ n_h \in X & \text{(ii)} \\ 2 \le n_h \le N_h & \text{(iii)} \end{array}\right\} j = 1, \ldots, p, \tag{7}$$

where $X = X_1$, or X_2, or X_3 is the feasible space of the problem and it is defined as

$$\begin{aligned} X_1 &= \left\{ n_h \in R^n \middle| \sum_{h=1}^{L} c_h n_h \le C, 2 \le n_h \le N_h, \text{ for } h = 1, 2, \ldots, L \right\}, \\ X_2 &= \left\{ n_h \in R^n \middle| \sum_{h=1}^{L} c_h n_h + \sum_{h=1}^{L} t_h \sqrt{n_h} \le C, 2 \le n_h \le N_h, \text{ for } h = 1, 2, \ldots, L \right\}, \\ X_3 &= \left\{ n_h \in R^n \middle| \sum_{h=1}^{L} c_h n_h + \sum_{h=1}^{L} t_h \sqrt{n_h} + \omega \sum_{h=1}^{L} \frac{n_h}{\lambda} \le C, 2 \le n_h \le N_h, \text{ for } h = 1, 2, \ldots, L \right\}. \end{aligned} \tag{8}$$

2.4. Weighted Goal Programming for Optimum Allocation Problem. In the goal programming approach, the p objective functions goals have been identified by solving the problem for individual j^{th} objective function ignoring the other $(j-1)$ objective functions with the feasible set constraints. The general form of goal programming is

$$\min_{n_h \in X} d\left(f(n_h), \widehat{f}\right), \tag{9}$$

where $\widehat{f} = (\widehat{f}_1, \widehat{f}_2, \ldots, \widehat{f}_p,)$ is the targeted goal vector which has been obtained at individual solutions and $d(f(n_h), \widehat{f})$ is the distance function between $d(f(n_h),$ and $\widehat{f}$ in some selected norm. The function (7) in the l_1 norm is given as

$$\min_{n_h \in X} d_1\left(f(n_h), \widehat{f}\right) = \sum_{j=1}^{p} \left| \sum_{h=1}^{L}\left(\frac{1}{n_h} - \frac{1}{N_h}\right)W_h^2 S_{jh}^2 - \widehat{f}_j \right|. \tag{10}$$

The weighted l_1 norm is

$$\min_{n_h \in X} d_1^{w}\left(f(n_h), \widehat{f}\right) = \sum_{j=1}^{p} w_j \left| \sum_{h=1}^{L}\left(\frac{1}{n_h} - \frac{1}{N_h}\right)W_h^2 S_{jh}^2 - \widehat{f}_j \right|, \tag{11}$$

where $w_j \ge 0$ is the weight assigned to the jth objective function.

The goal programming can be converted to a single-objective optimization problem by introducing the auxiliary variables

$$\begin{aligned} d_j^{+} &= \frac{\left\{ \left| \sum_{h=1}^{L} ((1/n_h) - (1/N_h))W_h^2 S_{jh}^2 - \widehat{f} \right|_j + \left(\sum_{h=1}^{L} ((1/n_h) - (1/N_h))W_h^2 S_{jh}^2 - \widehat{f}_j \right) \right\}}{2}, \\ d_j^{-} &= \frac{\left\{ \left| \sum_{h=1}^{L} ((1/n_h) - (1/N_h))W_h^2 S_{jh}^2 - \widehat{f}_j \right| - \left(\sum_{h=1}^{L} ((1/n_h) - (1/N_h))W_h^2 S_{jh}^2 - \widehat{f}_j \right) \right\}}{2}. \end{aligned} \tag{12}$$

Finally, the weighted goal programming model is

$$\left.\begin{array}{ll} \min \sum_{j=1}^{p} w_j\left(d_j^+ + d_j^-\right) & \text{(i)} \\ \text{subject to :} & \\ \sum_{h=1}^{L}\left(\frac{1}{n_h}-\frac{1}{N_h}\right)W_h^2 S_{jh}^2 - d_j^+ + d_j^- = \widehat{f}_j\left(n_h^*\right),\ \forall j & \text{(ii)} \\ n_h \in X & \text{(iii)} \\ d_j^+ \cdot d_j^- = 0,\ \forall j & \text{(iv)} \\ d_j^+ \geq 0, d_j^- \geq 0,\ \forall j & \text{(v)} \end{array}\right\}. \tag{13}$$

In (13) (i), the definition of d_j^+ & d_j^- is

$$d_j^+ = \begin{cases} \sum_{h=1}^{L}\left(\frac{1}{n_h}-\frac{1}{N_h}\right)W_h^2 S_{jh}^2 - \widehat{f}_j, \text{ if } \sum_{h=1}^{L}\left(\frac{1}{n_h}-\frac{1}{N_h}\right)W_h^2 S_{jh}^2 \geq \widehat{f}, \\ 0, \text{ if } \sum_{h=1}^{L}\left(\frac{1}{n_h}-\frac{1}{N_h}\right)W_h^2 S_{jh}^2 < \widehat{f}, \end{cases} \tag{14}$$

$$d_j^- = \begin{cases} \widehat{f} - \sum_{h=1}^{L}\left(\frac{1}{n_h}-\frac{1}{N_h}\right)W_h^2 S_{jh}^2, \text{ if } \widehat{f} \geq \sum_{h=1}^{L}\left(\frac{1}{n_h}-\frac{1}{N_h}\right)W_h^2 S_{jh}^2, \\ 0, if\ \widehat{f} < \sum_{h=1}^{L}\left(\frac{1}{n_h}-\frac{1}{N_h}\right)W_h^2 S_{jh}^2, \end{cases} \tag{15}$$

where d_j^+ and d_j^- are overachievement and underachievement functions respectively for the j^{th} goal value. It is further noted from (13) (iv) that d_j^+ and d_j^- can never be achieved simultaneously. It means that when overachievement is more significant than zero, then underachievement functions will be zero and vice versa. If the objective is maximization type function and hence underachievement function is not desirable. For this situation $w_j^+ = 0$ and $w_j^- = 1$, and equation (13) (i) objective function is reduced to $\min \sum_{j=1}^{p} w_j^- d_j^-$, where w_j^+ and w_j^- are the weight assigned to overachievement and underachievement functions, respectively. Conversely, for minimization type objective functions, the underachievement function is not desirable, that is, $w_j^+ = 0$ and $w_j^- = 1$, and (13) (i) objective function is reduced to $\min \sum_{j=1}^{p} w_j^+ d_j^+$.

3. Single-Objective Stochastic Multivariate Stratified Sampling Models

Deciding under uncertainty is challenging and unavoidable in most real-life problematic situations. The problems are mainly aim to optimize a set of function (s) under uncertain conditions by the decision-maker (s). If some or all of the constraints' parameters are unknown and are considered random, then such an optimization function becomes a stochastic programming problem.

Any modeling framework that optimized a problem under uncertainty can be viewed as a stochastic programming problem. The ultimate goal of these modeling types is to obtain a set of solution(s) that is feasible and optimal in some kind of set of data. Most of the models under this category involve parameters that follow probability distributions and can be known in advance or estimated using established procedures.

In general terms, stochastic programming can also be called probabilistic programming if some or all data of the optimization function follow probability distributions. In other words, variables that behave randomly in optimization problems can be regarded as stochastic or probabilistic as the case may be. Charnes & Cooper [33] developed and converted the chance-constrained programming technique into its equivalent deterministic nonlinear constraints.Many

authors have discussed the stochastic optimization problem. Among them are Prekopa [34], and Charnes and Cooper [35]. In the context of response surface methodology, Diaz Gracia et al. [36] had studied the problem under several stochastic optimization techniques. Diaz Gracia and Ramos-Quiroga [15, 16] formulated the problem of stratified sampling. In this problem, the authors have considered the sampling variances as a random variable. The sampling variances s_h^2 have an asymptotically normal distribution. The given problem converts into an equivalent deterministic problem by using a modified-E model.

$$\left.\begin{aligned}
&\min = k_1\left[\sum_{h=1}^{L}\frac{W_h^2}{(n_h-1)}s_h^2-\sum_{h=1}^{L}\frac{W_h^2}{N}\left(\frac{n_h}{n_h-1}\right)s_h^2\right]\\
&+k_2\left[\sum_{h=1}^{L}\frac{W_h^4}{n_h(n_h-1)^2}\left(C_{yh}^4-\left(s_h^2\right)^2\right)-\sum_{h=1}^{L}\frac{W_h^2}{N^2}\left(\frac{n_h}{(n_h-1)^2}\left(C_{yh}^4-\left(s_h^2\right)^2\right)\right)\right]^{1/2} \quad \text{(i)}\\
&\text{subject to :}\\
&\sum_{h=1}^{L}c_hn_h+c_o=C\,\text{(ii)}\\
&n_h\in N,\ h=1,2,\ldots,L.\,\text{(iii)}
\end{aligned}\right\}. \tag{16}$$

Again, Diaz Gracia and Garay Tapia [14] formulated the problem of stratified sampling. In this problem, authors have considered stochastic programming for minimizing the cost function under the constraint to a known bound for the estimated variance of the mean. The following problem converts into an equivalent deterministic problem by using chance constraints;

$$\left.\begin{aligned}
&\min = \sum_{h=1}^{L}c_hn_h+c_o && \text{(i)}\\
&\text{subject to :}\\
&\left[\sum_{h=1}^{L}\frac{W_h^2}{(n_h-1)}s_h^2-\sum_{h=1}^{L}\frac{W_h^2}{N}\left(\frac{n_h}{n_h-1}\right)s_h^2\right]+k_\alpha\\
&\left[\sum_{h=1}^{L}\frac{W_h^4}{n_h(n_h-1)^2}\left(C_{yh}^4-\left(s_h^2\right)^2\right)-\sum_{h=1}^{L}\frac{W_h^2}{N^2}\left(\frac{n_h}{(n_h-1)^2}\left(C_{yh}^4-\left(s_h^2\right)^2\right)\right)\right]^{1/2}\le V_o && \text{(ii)}\\
&n_h\in N,\ h=1,2,\ldots,L. && \text{(iii)}
\end{aligned}\right\}, \tag{17}$$

where V_o is a known non-negative constant and K_α is the value of the standard normal variable.

4. Multiobjective Stochastic Multivariate Stratified Sampling Models

In this section, we discussed the various nonlinear optimization sampling models under stochastic approaches. For instance, a problem of attaining several goals targets under probabilistic intervals was formulated as a linear stochastic model [37]. Problems involving stochastic MO have been analyzed considering different efficient concepts and establishing the relationships between the identified concepts [38]. A Multivariate Stratified random Sampling has been investigated where the asymptotic normality of the optimal solution was established, as well as the perturbation

effect of the stratum variance on the optimal solution [39]. Similarly, a problem of estimating several population means in an MSS design has been investigated [40]. The authors formulated an all-integer nonlinear model and proposed the solution using dynamic programming concepts with numerical illustrations. A multiobjective goal optimization in stratified sampling design was conducted by trading off between the sampling cost and its variance [41]. More than a single parameter estimation in a stratified sampling problem has been studied with a fixed budget and nonlinear cost [42]. Beale described the convex function minimization as a linear programming problem considering the coefficients as random variables [43].

Consider a multiobjective nonlinear programming problem (MNLPP) is

$$\left.\begin{array}{ll} \min f_j = \displaystyle\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right) & \text{(i)} \\ \text{subject to :} & \\ \displaystyle\sum_{h=1}^{L} c_n h_n + \sum_{h=1}^{L} t_n \sqrt{n_h} \le C & \text{(ii)} \\ 2 \le n_h \le N_h & \text{(iii)} \end{array}\right\} j = 1, \ldots, p. \tag{18}$$

Then, Equation (18) has been defined under the stochastic assumption given as the following stochastic nonlinear programming problem (SNLPP) for the p characteristics.

$$\left.\begin{array}{ll} \min = P\left(\displaystyle\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right) \le f_j\right) \ge \beta & \text{(i)} \\ \text{subject to :} & \\ P\left(\displaystyle\sum_{h=1}^{L} c_n h_n + \sum_{h=1}^{L} t_n \sqrt{n_h} \le C\right) \ge \beta & \text{(ii)} \\ 2 \le n_h \le N_h & \text{(iii)} \end{array}\right\} j = 1, \ldots, p. \tag{19}$$

Definition 1. A point is called feasible if and only if the probability measure of the event $g_j(n_h, \xi) \le 0$, $j = 1, \ldots, p$ is at least β. Or equivalently the constraints will be violated at most $(1 - \beta)$ times. The joint chance constraint is separately defined and is referred to as a separate chance constraint. That is,

$$P\left\{g_j(n_h, \xi) \le 0\right\} \ge \beta_j,\ j = 1, \ldots, p. \tag{20}$$

Now, by applying minimax chance constrained programming, (18) is as follows:

$$\left.\begin{array}{ll} \min\limits_{n_h} \max f_j & \text{(i)} \\ \text{subject to :} & \\ P\left(\displaystyle\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right) \le f_j\right) \ge \beta & \text{(ii)} \\ P\left(\displaystyle\sum_{h=1}^{L} c_n h_n + \sum_{h=1}^{L} t_n \sqrt{n_h} \le C\right) \ge \beta & \text{(iii)} \\ 2 \le n_h \le N_h\,(iv) & \end{array}\right\} j = 1, \ldots, p. \tag{21}$$

where β is the predetermined confidence level and $\min f_j$ is the variance term.

We can also formulate a stochastic goal programming for the problem defined in (21) with target goal values.

$$\left.\begin{array}{ll} \min\limits_{n_h} \displaystyle\sum_{j=1}^{p} \delta_j^+ & \text{(i)} \\ \text{subject to :} & \\ P\left(\displaystyle\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right) - \delta_j^+ \le f_j\right) \ge \beta & \text{(ii)} \\ P\left(\displaystyle\sum_{h=1}^{L} c_n h_n + \sum_{h=1}^{L} t_n \sqrt{n_h} \le C\right) \ge \beta & \text{(iii)} \\ 2 \le n_h \le N_h\,(iv) & \end{array}\right\} j = 1, \ldots, p. \tag{22}$$

Remark 1. (i) The stochastic objective constraints

$$P\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right) \le f_j\right) \ge \beta. \tag{23}$$

coincide with the form in (20) by defining

$$g_j(n_h, \xi) = \sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right) - f_j. \tag{24}$$

(ii) The stochastic goal constraints

$$P\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right) - f_j \ge \delta^+\right) \ge \beta. \tag{25}$$

coincide with the form in (20) by defining

$$g_j(n_h,\xi) = f_j - \sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right) + \delta^+. \quad (26)$$

where δ^+ is an overachievement goal.

(iii) The stochastic problem constraints

$$P\left(\sum_{h=1}^{L} c_n h_n + \sum_{h=1}^{L} t_n\sqrt{n_h} \le C\right) \ge \beta. \quad (27)$$

coincide with the form in (20) by defining

$$g_j(n_h,\xi) = \sum_{h=1}^{L} c_n h_n + \sum_{h=1}^{L} t_n\sqrt{n_h} - C. \quad (28)$$

(iv) The stochastic problem constraints

$$P\left(\sum_{h=1}^{L} c_n h_n + \sum_{h=1}^{L} t_n\sqrt{n_h} \le C\right) \ge \beta. \quad (29)$$

coincide with the form in (20) by defining

$$g_j(n_h,\xi) = C - \sum_{h=1}^{L} c_n h_n + \sum_{h=1}^{L} t_n\sqrt{n_h} + \delta^+. \quad (30)$$

where overachievement goal.

(v) For a continuous random variable ξ, the value $Pk_\beta \le \xi = 1 - \Phi(k_\beta)$ holds always, and we have $k_\beta = \Phi^{-1}(1-\beta)$, where Φ^{-1} is the inverse function of Φ.

4.1. Conversion of Stochastic Inequalities to Equivalent Deterministic. In (18) (i), the term s_{jh}^2 is assumed to be a random variable. In practice, some approximations of these parameters, which are known from some preliminary or recent survey, may be used. The concept of limiting the distribution of the sample variances in a sampling problem is used in [39], considering the random variable ξ_h defined as

$$\xi = \frac{1}{n_h - 1}\sum_{i=1}^{n_h}\left(y_{jhi} - \overline{Y}_{jh}\right)^2, \quad (31)$$

where $\overline{Y}_{jh} = (1/N_h)\sum_{i=1}^{N_h} y_{jhi}$. Note that ξ_h has an asymptotically normal distribution with mean

$$E(\xi_h) = \frac{n_h}{n_h - 1} S_{jh}^2, \quad (32)$$

and variance

$$V(\xi_h) = \frac{n_h}{(n_h - 1)^2}\left[C_{jh}^4 - \left(S_{jh}^2\right)^2\right], \quad (33)$$

respectively, where

$$C_{jh}^4 = \frac{1}{N_h}\sum_{i=1}^{N_h}\left(y_{jhi} - \overline{Y}\right)^4, \quad (34)$$

is the fourth central moment of j^{th} character in the h^{th} stratum. The sequence of sample variances is given by

$$S_{jh}^2 = \xi_h - \frac{n_h}{N_h - 1}\sum_{i=1}^{N_h}\left(\overline{y}_{jhi} - \overline{Y}_{jh}\right)^2, \quad (35)$$

where $n_h/N_h - 1 \longrightarrow 1$ and $(\overline{y}_{jhi} - \overline{Y}_{jh})^2 \longrightarrow 0$ in probability and hence under the asymptotically normal property $S_{jh}^2 \longrightarrow^a N(E(\xi), \mathrm{Var}(\xi))$, $h = 1, 2, \ldots, L$ are independents.

Based on the above discussion, the multivariate stratified sampling variance function with the following expected function and variance function

$$\left.\begin{aligned} E\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right) &= \sum_{h=1}^{L}\left(\frac{W_h^2 E(\xi_h)}{n_h} - \frac{W_h^2 E(\xi_h)}{N_h}\right) \\ &= \sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h - 1}\right) - \left(\frac{W_h^2 S_{jh}^2}{N_h}\right)\frac{n_h}{n_h - 1} \cong \sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h - 1}\right) - \left(\frac{W_h^2 S_{jh}^2}{N_h}\right) \end{aligned}\right\}, \quad (36)$$

as n_h is sufficiently large,

$$\left.\begin{aligned} V\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right) &= \sum_{h=1}^{L}\left(\frac{W_h^4 V(\xi_h)}{n_h^2} - \frac{W_h^4 V(\xi_h)}{N_h^2}\right) \\ &= \sum_{h=1}^{L}\left(\frac{W_h^4\left(C_{jh}^4 - \left(S_{jh}^2\right)^2\right)}{n_h(n_h - 1)}\right) - \left(\frac{W_h^4\left(C_{jh}^4 - \left(S_{jh}^2\right)^2\right)}{N_h^2}\right)\frac{n_h}{(n_h - 1)^2} \\ &\cong \sum_{h=1}^{L}\left(\frac{W_h^4\left(C_{jh}^4 - \left(S_{jh}^2\right)^2\right)}{n_h(n_h - 1)}\right) - \left(\frac{W_h^4\left(C_{jh}^4 - \left(S_{jh}^2\right)^2\right)}{N_h^2}\right) \end{aligned}\right\}, \quad (37)$$

and as n_h is sufficiently large.

Theorem 1. *Assume that the stochastic vector ζ degenerates to a random variable ξ with a probability distribution Φ, and the function $g_j(n_h, \zeta)$ has the form $g_j(n_h, \zeta) = \overline{\omega}_j(n_h) - \xi$. Then, $P\{g_j(n_h, \zeta)\} \le 0 \ge \beta$ iff $\overline{\omega}_j(n_h) \le k_\beta$, where k_β is the maximal value such that $P\{k_\beta \le \xi\} \ge \beta$. Note that the probability $P\{k_\beta \le \xi\}$ will be increased if k_β is replaced with a smaller number.*

$$E\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right) + \Phi^{-1}(\widehat{\beta})\sqrt{V\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right)} \le f_j, \tag{39}$$

where Φ is the standardized normal distribution function.

Proof. In the probability model of survey sampling, the probability that the sampling variance value is smaller or equal to an absolute goal value is maximized. That is,

$$P\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right) \le f_j^*, \tag{40}$$

$$\begin{aligned}
&\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right) \le f_j^* \\
&= P\left(\frac{f_j(n_h) - E\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}{\sqrt{V\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}} \le \frac{f_j^* - E\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}{\sqrt{V\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}}\right) \\
&= P\left(\eta \le \frac{f_j^* - E\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}{\sqrt{V\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}}\right) = \Phi\left(\frac{f_j^* - E\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}{\sqrt{V\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}}\right),
\end{aligned} \tag{41}$$

where η is the standardized normally distributed random variable.

From the fact that

$$\Phi\left(\frac{f_j^* - E\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}{\sqrt{V\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}}\right) \ge \beta, \tag{42}$$

it is equivalent to

Theorem 2. *Assume that the stochastic function and $g(n_h, \xi)$ has the form*

$$g_j(n_h, \xi) = \sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right) - f_j. \tag{38}$$

If s_{jh}^2 are assumed to be independent normally distributed random variables, then $\{g(n_h, \xi) \le 0\} \ge \beta$ if and only if

where f_j^* is the minimum target goal value for the jth objective function.

Recall the (18) (i), independently normally distributed random variables s_{jh}^2. Moreover, covariance terms will be zero, and only the variance terms will be remain.

$$\begin{aligned}
&\Phi^{-1}\beta\sqrt{V\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right)} \le f_j^* \\
&- E\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right),
\end{aligned} \tag{43}$$

and from (43), the maximum of β is searched in the interval $(0, 1)$. The following function is defined for convenience:

$$f_o(n_h; \beta) = E\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right) + \Phi^{-1}\beta\sqrt{V\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right)} - f_j^*. \tag{44}$$

Here, we assume that for an optimal solution (n_h^*, β^*) to the problem (44), $\beta^* > 0.5$ holds. From this assumption, $\Phi^{-1}(\beta) > 0$, and then for a fixed value of β^*, it follows that $f_o(n_h^*; \beta^*)$ is convex. □

Proposition 1. *Let (n_h^*, β^*) be an optimal solution to the problem (21) with a target value f_j larger than*

$$f_{oj} = E\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right), \tag{45}$$

i.e., $f_j > f_{oj}$. Then, $\beta^* > 0.5$ holds.

Proof. From the condition $f_j > f_{oj} > 0$, of the preposition, $f_j - f_{oj} > 0$, and then we have

$$\Phi\left(\frac{f_j^* - E\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}{\sqrt{V\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}}\right) > 0.5. \tag{46}$$

Letting

$$\widehat{\beta} = \Phi\left(\frac{f_j^* - E\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}{\sqrt{V\left(\sum_{h=1}^{L}\left(\left(W_h^2 S_{jh}^2/n_h\right) - W_h^2 S_{jh}^2/N_h\right)\right)}}\right), \tag{47}$$

we have

$$E\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right) + \Phi^{-1}\beta\sqrt{V\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right)} = f_j. \tag{48}$$

Therefore, $(\widehat{n}_h, \widehat{\beta})$ is a feasible solution to the problem (44) with the target value f_j. Since the value (n_h^*, β^*) is the optimal solution of (44), it holds.

Theorem 3. *Assume that the stochastic vector $\xi = (c_1, c_2, \ldots, c_L, t_1, t_2, \ldots, t_L, C)$ and the function $g(n_h, \xi)$ has the form $g(n_h, \xi) = \sum_{h=1} L c_h n_h + \sum_{h=1}^{L} t_h\sqrt{n_h} - C$. If c_i and C are assumed to be independent normally distributed random variables, then $\{g(n_h, \xi) \le 0\} \ge \beta$ if and only if*

$$\Phi^{-1}\beta \le -\left(\frac{\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} - E(C)}{\sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h + V(C)}}\right), \tag{49}$$

or equivalently

$$\frac{\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} + \Phi^{-1}(\beta)}{\sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h + (C)}} \le E(C), \tag{50}$$

where Φ is the standardized normal distribution function.

Proof. Let chance constraint, that is,

$$P\left(\sum_{h=1}^{L} c_h n_h + \sum_{h=1}^{L} t_h\sqrt{n_h} \le C\right) \ge \beta. \tag{51}$$

It is assumed that C, c_h, and t_h are normally distributed random variables. Moreover, assume that all are independent of each other. We note that

$$\frac{g_j(n_h, \xi) - \left(\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} - E(C)\right)}{\sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h + V(C)}}. \tag{52}$$

Equation (52) is the standard normal random variable $N(0, 1)$, and it follows

$$P\left(\sum_{h=1}^{L} c_h n_h + \sum_{h=1}^{L} t_h\sqrt{n_h} \le C\right)$$

$$= P\left(\frac{g_j(n_h, \xi) - \left(\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} - E(C)\right)}{\sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h + V(C)}}\right.$$

$$\left. \le -\frac{\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} - E(C)}{\sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h + V(C)}}\right), \tag{53}$$

or equivalent to

$$P\left(\eta \le -\frac{\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} - E(C)}{\sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h + V(C)}}\right) \ge \beta, \tag{54}$$

where η is the standardized, normally distributed random variable. The above-given constraint holds if and only if

$$\Phi^{-1}\beta \le -\left(\frac{\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} - E(C)}{\sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h + V(C)}}\right). \tag{55}$$

Hence, the chance constraint (12) (ii) can be transformed into

$$\frac{\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} + \Phi^{-1}(\beta)}{\sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h + V(C)}} \le E(C). \tag{56}$$

It can be further assumed that only c_h and t_h are normally distributed random variables and independent to each other where the total budget for the survey is fixed. The same procedure as discussed above will be followed and hence the cost constraint defined in (56) will be defined as follows:

$$\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} + \Phi^{-1}(\beta) \sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h} \le C. \tag{57}$$

□

$$\min_{n_h} = \sum_{j=1}^{p} \delta_j^+, \text{ (i)}$$

Subject to :

$$E\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right) + \Phi^{-1}(\beta)\sqrt{V\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right)} - \delta_j^+ \le f_j^* \text{ (ii)} \tag{58}$$

$$\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} + \Phi^{-1}(\beta)\sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h} \le C \text{ (iii)}$$

$2 \le n_h \le N_h$ and $\delta_J^+ \ge 0$ (iv).

The individual sampling variance goal value can be obtained using the following define equation:

$$\min_{j=1,2,\ldots,p} f_j = E\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right) + \Phi^{-1}(\beta)\sqrt{V\left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{jh}^2}{n_h} - \frac{W_h^2 S_{jh}^2}{N_h}\right)\right)},$$

Subject to :

$$\sum_{h=1}^{L} E(c_h) n_h + \sum_{h=1}^{L} E(t_h)\sqrt{n_h} + \Phi^{-1}(\beta)\sqrt{\sum_{h=1}^{L} V(c_h) n_h^2 + \sum_{h=1}^{L} V(t_h) n_h} \le C \tag{59}$$

$2 \le n_h \le N_h$.

4.3. Chebychev Goal Programming Sampling Model. In this method, first, we set goals for each objective that we want to attain. Let some goals $g = (g_1, g_2, \ldots, g_k)'$ be identified for the objective functions $f = (f_1(x_h), f_2(x_h), \ldots, f_k(x_h))'$. Let the objective functions $f = (f_1^*(x_h), f_2^*(x_h), \ldots f_k^*(x_h))'$ be defined as close as possible to goals $g = (g_1, g_2, \ldots, g_k)'$.

The difference between $f = (f_1^*(x_h), f_2^*(x_h), \ldots, f_k^*(x_h))'$ and $g = (g_1, g_2, \ldots, g_k)'$ is defined as the deviation function $D(f(n_h), g)$. In a sampling optimization problem, the aim is to find an $n_h^* \in X$, which minimizes $D(f(n_h), g)$, that is,

$$n_h^* = \arg\min_{n_h \in X} D(f(n_h), g), \tag{60}$$

where

$$D(f(n_h), g) = \max D\{f_1(n_h), g_1), \ldots, f_k(n_h), g_k)\}, \tag{61}$$

is the maximum deviation of individual goals. Finally, a preferred solution is then defined as one that minimizes the maximum deviations from the goals. In light of the above discussion, the problem formulated in (55) is transformed equivalently with an auxiliary variable δ as follows:

4.2. Stochastic Goal Programming Sampling Variance Model. In light of the above discussion, the problem formulated in (57) is transformed equivalently as follows:

$$\left.\begin{array}{l} \min\limits_{n_h} \delta, \quad \text{(i)} \\ \text{subject to :} \\ E\left(\sum\limits_{h=1}^{L} \dfrac{W_h^2 S_{jh}^2}{n_h} - \dfrac{W_h^2 S_{jh}^2}{N_h}\right) + \Phi^{-1}\beta\sqrt{V\left(\sum\limits_{h=1}^{L} \dfrac{W_h^2 S_{jh}^2}{n_h} - \dfrac{W_h^2 S_{jh}^2}{N_h}\right)} \quad \text{(ii)} \\ \sum\limits_{h=1}^{L} E(t_h)\sqrt{n_h} + \Phi^{-1}\beta\sqrt{\sum\limits_{h=1}^{L} V(c_h) n_h^2 + \sum\limits_{h=1}^{L} V(t_h) n_h} - C \le \delta \quad \text{(iii)} \\ 2 \le n_h \le N_h \delta \text{ and } \delta \ge 0 \quad (iv) \end{array}\right\}. \tag{62}$$

4.4. Stochastic Sampling Cost Model. Similar to the sampling variance model, in a sampling cost model, we minimize the objective cost function subject to variance constraints. That is,

$$\left.\begin{array}{l} \min\limits_{n_h} = \sum\limits_{h=1}^{L} E(c_h) n_h + \sum\limits_{h=1}^{L} E(t_h)\sqrt{n_h} + \Phi^{-1}(\beta)\sqrt{V\left(\sum\limits_{h=1}^{L}\left(\dfrac{W_h^2 S_{jh}^2}{n_h} - \dfrac{W_h^2 S_{jh}^2}{N_h}\right)\right)} \\ \text{Subject to :} \\ E\left(\sum\limits_{h=1}^{L}\left(\dfrac{W_h^2 S_{jh}^2}{n_h} - \dfrac{W_h^2 S_{jh}^2}{N_h}\right)\right) + \Phi^{-1}(\beta)\sqrt{V\left(\sum\limits_{h=1}^{L}\left(\dfrac{W_h^2 S_{jh}^2}{n_h} - \dfrac{W_h^2 S_{jh}^2}{N_h}\right)\right)} \le f_j^* \\ 2 \le n_h \le N_h \text{ and } j = 1, 2, \ldots, p. \end{array}\right\}, \tag{63}$$

where C^* is the target goal value.

5. Linear Approximation of Multivariate Stratified Sampling Problem

The objective function in (7) f_j is linearized at the individual optimum points [44]. Thus, for $j = q$ at the point $\underline{n}_q^* = (n_{q1}^*, n_{q2}^*, \ldots, n_{qh}^*)$, f_q' may be approximated by the linear function with n_h as

$$f_q' \cong f_q\left(n_{qh}^*\right) + \nabla' f_q\left(n_{qh}^*\right)\left(n_h - n_{qh}^*\right), \tag{64}$$

where f_q' is the variation term and $\nabla' f_q(n_{qh}^*)$ is the value of the vector of partial derivatives of f_q with respect to n_{qh} $(h = 1, 2, \ldots, L)$ at the point n_{qh}^* given as follows:

$$\nabla' f_q'\left(n_{qh}^*\right) = \left[-\left(\frac{W_1^2 S_{q1}^2}{\left(n_{q1}^*\right)^2}\right), -\left(\frac{W_2^2 S_{q2}^2}{\left(n_{q2}^*\right)^2}\right), \ldots, -\left(\frac{W_L^2 S_{qL}^2}{\left(n_{qL}^*\right)^2}\right)\right]. \tag{65}$$

This gives

$$\begin{aligned} &\nabla' f_q'\left(n_{qh}^*\right)\left(n_h - n_{qh}^*\right) \\ &= \left\{\left(\sum_{h=1}^{L} \frac{W_h^2 S_{qh}^2}{n_{qh}^*}\right) - \left(\sum_{h=1}^{L}\left(\frac{W_h^2 S_{qh}^2}{n_{qh}^*}\right) n_h\right)\right\}, \\ &f_q' \cong 2\left\{\sum_{h=1}^{L} \frac{W_h^2 S_{qh}^2}{n_{qh}^*}\right\} - \left\{\sum_{h=1}^{L}\left(\frac{W_h^2 S_{qh}^2}{n_{qh}^*}\right) n_h\right\} = f_q'(\text{say}). \end{aligned} \tag{66}$$

After dropping the constant terms in the linear objective function, the NLPP (6) can be approximated, and the final problem is equivalent to maximizing $(-f_k')$. That is,

$$\left.\begin{array}{l} \max f_j = \sum\limits_{h=1}^{L}\left(\dfrac{W_h^2 S_{jh}^2}{\left(n_{jh}^*\right)^2}\right) n_h, \text{ (i)} \\ \text{subject to :} \\ \sum\limits_{h=1}^{L} c_n h_n \le C \quad \text{(ii)} \\ 2 \le n_h \le N_h \quad \text{(iii)} \end{array}\right\} j = 1, 2, \ldots, p. \tag{67}$$

Table 1: Data for two characteristics and four strata.

h	N_h	S_{h1}^2	S_{h2}^2	W_h	c_h
1	174	268535.9	136457.3	0.11233	9
2	196	322381.3	146465.2	0.12653	8
3	676	355325.6	127272	0.43641	12
4	503	324026.4	141460.7	0.32473	11

Table 2: Data for two characteristics and four strata.

h	N_h	W_h	S_{h1}^2	S_{h2}^2	C_{h1}^4	C_{h2}^4	$E(c_h)$	$V(c_h)$	$E(t_h)$	$V(t_h)$
1	174	0.11233	21.12788	6.393020	1812.391	106.4593	9	2.10	2.00	0.35
2	196	0.12653	24.56889	5.326552	1635.366	66.60126	8	1.75	2.45	0.75
3	676	0.43641	22.81545	6.220776	1517.975	112.4234	12	2.50	2.30	0.60
4	503	0.32473	23.37032	6.076825	1750.445	106.6261	11	2.25	2.10	0.55

6. Numerical Results

This section presents some numerical examples to illustrate the various theoretical concepts discussed above.

Example 1. A simulation study is used to show the computational procedure for the theoretical discussion of multiobjective MSS. The *R* package (stratifyR) [45] is used to simulate the data for the two different characteristics, which are divided into four strata. The information on simulation studies is given in Table 1.

The available budget for the survey is $C_0 = \$2500$.

Using (7), in Section 2.3, the best individual optimal solutions for both characteristic $j = 1, 2$ are obtained as follows:

$f_1 = 1218.183, n_1 = 26, n_2 = 33, n_3 = 99, n_4 = 74$ and $.f_2 = 494.3353, n_1 = 29, n_2 = 36, n_3 = 92, n_4 = 77.$ The Weighted Goal Programming discussed in 2.4 is applied to obtain the compromised allocations using (13) with the help of the LINGO optimization package [46], as follows: $f_1 = 1219.624$, $f_2 = 496.7824, n_1 = 27, n_2 = 35, n_3 = 96, n_4 = 75$.

Example 2. A simulation study is used to show the computational procedure of the stochastic multiobjective multivariate stratified sampling. The *R* package (stratifyR) [45] is used to simulate the data for the two different characteristics, which are divided into four strata. The information on simulation studies is given in Table 2.

The available budget for the survey is $C_0 = \$2000$.

The calculated parameters used in this study are presented in Table 3.

The individual solutions of numerical Example 2 for both characteristics $j = 1, 2$ are obtained using (59) of Section 4.2 as follows:

$f_1 = 0.1517723, n_1 = 20, n_2 = 26, n_3 = 64, n_4 = 52.$ and $f_2 = 0.3982894, n_1 = 19, n_2 = 22, n_3 = 67, n_4 = 52$. The stochastic Goal programming discussed in Sec. 4.2 is applied to obtain the compromised allocations using (58) as follows: $f_1 = 0.03982894$, $f_2 = 0.03982894, n_1 = 19, n_2 = 22, n_3 = 67, n_4 = 52$. The Chebychev Goal programming discussed in Section 4.3 is applied to obtain the compromised allocations using (62) as follows: $f_1 = 0.03982894, f_2 = 0.03982894, n_1 = 19, n_2 = 22, n_3 = 67, n_4 = 52$. The Stochastic Sampling Cost Model discussed in Section 4.4 is applied to obtain the compromised allocations using (63) as follows: $C = 1997.716, n_1 = 21, n_2 = 24, n_3 = 67, n_4 = 49$.

Example 3. Here, the linearization of sampling variance are discussed numerically. Using the data of Table 1 in (67), the following sample allocations for $j = 1$ are $n_{11} = 2, n_{12} = 195, n_{13} = 75, n_{14} = 2$ are obtained with the sampling variance of $f_{\text{linear}} = \sum_{h=1}^{L}((1/n_h) - (1/N_h))W_h^2 S_{jh}^2 = 1465.552$. Solving the nonlinear sampling variance problem defined in (7) with the same data, the sample allocation was perceived to be $n_{11} = 26, n_{12} = 33, n_{13} = 99, n_{14} = 74$ with sampling variance of $f_{\text{nonlinear}} = 1218.183$. It is observed that the sample allocation from the nonlinear problem is better than the linearized one. Also, the sampling variability is higher in the linearized model than in the nonlinear. Therefore, it can be concluded that linearization of the sampling problem does not give better sample allocations as well as minimum sample variance.

Example 4. Here, the linear approximation of sampling variance are numerically presented. Using the data of Table 1 in (67), the following sample allocations for $j = 2$ are $n_{21} = 2, n_{22} = 3, n_{23} = 203, n_{24} = 2$ are obtained with the sampling variance $f_{\text{linear}} = \sum_{h=1}^{L}(1/n_h - 1/N_h)W_h^2 S_{jh}^2 = 595.91$. Solving the nonlinear sampling variance problem with the same data, the sample allocation was perceived to be $n_{21} = 29, n_{22} = 36, n_{23} = 92, n_{24} = 77$ with sampling variance of $f_{\text{nonlinear}} = 494.3353$. It is again observed that the sample allocation from the nonlinear problem is better than the linearized one as in the case of Example 3. Also, the sampling variability is higher in the linearized model than in the nonlinear. Therefore, it can be concluded that linearization of the sampling problem does not give optimal sample allocations as well as minimum sample variance.

In general, It can be concluded that linearization of nonlinear sampling variance in a survey sampling problem does not help to determine the optimal sample allocations with minimum variability since approximation of nonlinear into a linear function will not sufficiently optimize the function value as a result of a loss of generality during the

Table 3: Data for two characteristics and four strata.

$W_h^2 S_{1h}^2$	$W_h^2 S_{2h}^2$	$C_{1h}^4 - (S_{h1}^2)^2$	$W_h^4 (C_{1h}^4 - (S_{h1}^2)^2)$	$C_{2h}^4 - (S_{h2}^2)^2$	$W_h^4 (C_{2h}^4 - (S_{h2}^2)^2)$
0.2665922	0.08066731	1366.004	0.2174879	65.5886	0.01044267
0.3933440	0.08527725	1031.736	0.2644493	38.2291	0.009798693
4.345287	1.184770	997.4302	36.17940	75.4956	2.738422
2.46439	0.6407986	1204.273	13.39105	69.6983	0.7750179

linear transformation of sampling variance. In such cases, it has been observed that the optimal global solution (Pareto optimal solution) of a function can suffer. Suppose if the original nonlinear function is convex and we approximate it into the linear function, we know that the linear function can be convex or concave. Indeed, the transformation process of nonlinear to a linear function could compromise several properties of the nonlinear based on the fact that a nonlinear function has a high convergence rate to a linear function. Hence, it can be verified that the linearized sampling variance case optimal allocation has high variability compared to the actual sampling problem. Hence, it can be concluded that the need for linearizing the sampling variance function for obtaining the optimal sample allocation is not an optimal decision in a sampling survey. Since the linearization of sampling design under deterministic does not give an efficient solution, there is no need to carry out the same under stochastic, and therefore, the study did not consider linearization under the uncertainty. Other interested researchers can explore the context of different sampling designs.

7. Conclusion

In sample design, allocating samples efficiently and attaining maximum accuracy in minimizing variances plays an important role. Various techniques, theorems, properties and prepositions, and stochastic models were studied, discussed, and presented for the multiobjective multivariate stratified sampling scheme. The discussion is supported with numerical examples in each case. This research is a theoretical framework and conceptual methodology for survey sampling in optimal allocation problems in a certain and stochastic environment. Based on the discussion and numerical illustrations, it can be deduced that sampling variance values resulting from the linearization have higher variability than the nonlinear sampling variance case. Therefore, the study suggests that there is no need for linearizing the original sampling variance function with the hope of getting an optimal decision regarding sampling allocation in survey sampling. The interested researchers can further demonstrate the usefulness and power of the techniques and methods presented in this study. In the future, the study could be extended to more sampling designs in optimal allocation problems for survey sampling.

Data Availability

Not applicable.

Conflicts of Interest

The authors declare that there are no known conflicts of interest regarding financial or authorship arrangement for this research.

Authors' Contributions

The authors contributed equally to the paper.

Acknowledgments

This research was funded by the Deanship of Scientific Research of Jazan University, under the Waed grant number w41-54.

References

[1] P. V. Sukhatme, B. V. Sukhatme, S. Sukhatme, and C. Asok, *Sampling Theory of Surveys with Application*, pp. 21–25, Indian society of agricultural statistics and IOWA State University Press Ames, India and USA, 1984.

[2] M. G. Khan, T. Maiti, and M. J. Ahsan, "An optimal multivariate stratified sampling design using auxiliary information: an integer solution using goal programming approach," *Journal of Official Statistics*, vol. 26, no. 4, p. 695, 2010.

[3] R. Varshney, M. G. M. KhanKhan, U. Fatima, and M. J. Ahsan, "Integer compromise allocation in multivariate stratified surveys," *Annals of Operations Research*, vol. 226, no. 1, pp. 659–668, 2015.

[4] S. Ghufran, S. Khowaja, and M. J. Ahsan, "Optimum multivariate stratified sampling designs with travel cost: a multiobjective integer nonlinear programming approach," *Communications in Statistics - Simulation and Computation*, vol. 41, no. 5, pp. 598–610, 2012.

[5] N. Gupta and A. Bari, "Fuzzy multi-objective optimization for optimum allocation in multivariate stratified sampling with quadratic cost and parabolic fuzzy numbers," *Journal of Statistical Computation and Simulation*, vol. 87, no. 12, pp. 2372–2383, 2017.

[6] Y. S. Raghav, I. Ali, and A. Bari, "Multi-objective nonlinear programming problem approach in multivariate stratified sample surveys in the case of non-response," *Journal of Statistical Computation and Simulation*, vol. 84, no. 1, pp. 22–36, 2014.

[7] S. Khowaja, S. Ghufran, and M. J. Ahsan, "Estimation of population means in multivariate stratified random sampling," *Communications in Statistics - Simulation and Computation*, vol. 40, no. 5, pp. 710–718, 2011.

[8] I. Ali, Y. S. Raghav, and A. Bari, "Compromise allocation in multivariate stratified surveys with stochastic quadratic cost function," *Journal of Statistical Computation and Simulation*, vol. 83, no. 5, pp. 962–976, 2013.

[9] R. Varshney, "Optimum allocation in multivariate stratified sampling design in the presence of nonresponse with Gamma cost function," *Journal of Statistical Computation and Simulation*, vol. 89, no. 13, pp. 2454–2467, 2019.

[10] K. G. Reddy and M. G. M. Khan, "Optimal stratification in stratified designs using weibull-distributed auxiliary information," *Communications in Statistics - Theory and Methods*, vol. 48, no. 12, pp. 3136–3152, 2019.

[11] R. Varshney, S. Gupta, and I. Ali, "An optimum multivariate-multiobjective stratified sampling design: fuzzy programming approach," *Pakistan Journal of Statistics and Operation Research*, vol. 13, no. 4, pp. 829–855, 2017.

[12] S. Ghufran, S. Khowaja, and M. J. Ahsan, "Compromise allocation in multivariate stratified sample surveys under two stage randomized response model," *Optimization Letters*, vol. 8, no. 1, pp. 343–357, 2014.

[13] S. Khowaja, S. Ghufran, and M. J. Ahsan, "On the problem of compromise allocation in multi-response stratified sample surveys," *Communications in Statistics - Simulation and Computation*, vol. 42, no. 4, pp. 790–799, 2013.

[14] J. A. Díaz-García and M. M. Garay-Tápia, "Optimum allocation in stratified surveys: stochastic programming," *Computational Statistics & Data Analysis*, vol. 51, no. 6, pp. 3016–3026, 2007.

[15] J. A. Díaz-García and R. Ramos-Quiroga, "Optimum allocation in multivariate stratified random sampling: a modified prékopa's approach," *Journal of Mathematical Modelling and Algorithms in Operations Research*, vol. 13, no. 3, pp. 315–330, 2014.

[16] J. A. Díaz-García and R. Ramos-Quiroga, "Stochastic optimal design in multivariate stratified sampling," *Optimization*, vol. 63, no. 11, pp. 1665–1688, 2014.

[17] A. Haq, I. Ali, and R. Varshney, "Compromise allocation problem in multivariate stratified sampling with flexible fuzzy goals," *Journal of Statistical Computation and Simulation*, vol. 90, no. 9, pp. 1557–1569, 2020.

[18] M. G. M. KhanKhan, M. J. Ahsan, and N. Jahan, "Compromise allocation in multivariate stratified sampling: an integer solution," *Naval Research Logistics*, vol. 44, no. 1, pp. 69–79, 1997.

[19] E. B. Tirkolaee, A. Goli, P. Ghasemi, and F. Goodarzian, "Designing a sustainable closed-loop supply chain network of face masks during the COVID-19 pandemic: pareto-based algorithms," *Journal of Cleaner Production*, vol. 333, Article ID 130056, 2022.

[20] A. Goli and H. Mohammadi, "Developing a Sustainable Operational Management System Using Hybrid Shapley Value and Multimoora Method: Case Study Petrochemical Supply Chain," *Environment, Development and Sustainability*, pp. 1–30, 2021.

[21] A. Goli and B. Malmir, "A covering tour approach for disaster relief locating and routing with fuzzy demand," *International Journal of Intelligent Transportation Systems Research*, vol. 18, no. 1, pp. 140–152, 2020.

[22] A. Goli and T. Keshavarz, "Just-in-time scheduling in identical parallel machine sequence-dependent group scheduling problem," *Journal of Industrial and Management Optimization*, 2021.

[23] A. Goli, H. Khademi ZareKhademi Zare, R. Tavakkoli-Moghaddam, and A. Sadeghieh, "Hybrid artificial intelligence and robust optimization for a multi-objective product portfolio problem Case study: the dairy products industry," *Computers & Industrial Engineering*, vol. 137, Article ID 106090, 2019.

[24] M. Alinaghian and A. Goli, "Location, allocation and routing of temporary health centers in rural areas in crisis, solved by improved harmony search algorithm," *International Journal of Computational Intelligence Systems*, vol. 10, no. 1, pp. 894–913, 2017.

[25] A. Goli, H. Khademi-Zare, R. Tavakkoli-Moghaddam, A. Sadeghieh, M. Sasanian, and R. Malekalipour Kordestanizadeh, "An integrated approach based on artificial intelligence and novel meta-heuristic algorithms to predict demand for dairy products: a case study," *Network: Computation in Neural Systems*, vol. 32, no. 1, pp. 1–35, 2021.

[26] I. Ali, U. M. Modibbo, J. Chauhan, and M. Meraj, "An integrated multi-objective optimization modelling for sustainable development goals of India," *Environment, Development and Sustainability*, vol. 23, no. 3, pp. 3811–3831, 2021.

[27] U. M. Modibbo, I. Ali, and A. Ahmed, "Multi-objective optimization modelling for analysing sustainable development goals of Nigeria: agenda 2030," *Environment, Development and Sustainability*, vol. 23, no. 6, pp. 9529–9563, 2021.

[28] A. AlArjani, U. M. Modibbo, I. Ali, and B. Sarkar, "A new framework for the sustainable development goals of Saudi Arabia," *Journal of King Saud University Science*, vol. 33, no. 6, Article ID 101477, 2021.

[29] M. Barma and U. M. Modibbo, "Multiobjective mathematical optimization model for municipal solid waste management with economic analysis of reuse/recycling recovered waste materials," *Journal of Computational and Cognitive Engineering*, 2022.

[30] Y. S. Muhammad, J. Shabbir, I. Husain, and M. Abd-el.Moemen, "Multi-objective compromise allocation in multivariate stratified sampling using extended lexicographic goal programming with gamma cost function," *Journal of Mathematical Modelling and Algorithms in Operations Research*, vol. 14, no. 3, pp. 255–265, 2015.

[31] J. Beardwood, J. H. Halton, and J. M. Hammersley, "The shortest path through many points," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 55, no. 4, pp. 299–327, 1959.

[32] J. Neyman, "On the two different aspects of the representative method: the method of stratified sampling and the method of purposive selection," *Journal of the Royal Statistical Society*, vol. 97, no. 4, pp. 558–625, 1934.

[33] A. Charnes and W. W. Cooper, "Chance-constrained programming," *Management Science*, vol. 6, no. 1, pp. 73–79, 1959.

[34] A. Prekopa, "On probabilistic constrained programming," *Mathematical Programming Study*, vol. 28, pp. 113–138, 1970.

[35] A. Charnes and W. W. Cooper, "Deterministic equivalents for optimizing and satisficing under chance constraints," *Operations Research*, vol. 11, no. 1, pp. 18–39, 1963.

[36] J. A. Díaz-García, R. Ramos-Quiroga, and E. Cabrera-Vicencio, "Stochastic programming methods in the response surface methodology," *Computational Statistics & Data Analysis*, vol. 49, no. 3, pp. 837–848, 2005.

[37] B. Contini, "A stochastic approach to goal programming," *Operations Research*, vol. 16, no. 3, pp. 576–586, 1968.

[38] A. M. Geoffrion, "Stochastic programming with aspiration or fractile criteria," *Management Science*, vol. 13, no. 9, pp. 672–679, 1967.

[39] A. Melaku, "Asymptotic normality of the optimal allocation in multivariate stratified random sampling," *Sankhya: The Indian Journal of Statistics, Series B*, vol. 48, pp. 224–232, 1986.

[40] M. G. Khan, E. A. Khan, and M. J. Ahsan, "An optimal multivariate stratified sampling design using dynamic programming," *Australian & New Zealand Journal of Statistics*, vol. 45, no. 1, pp. 107–113, 2003.

[41] M. Yousaf Shad and I. Husain, "Trade-off between cost and variance for a multi-objective compromise allocation in stratified random sampling," *Communications in Statistics - Theory and Methods*, vol. 46, no. 6, pp. 2655–2666, 2017.

[42] R. Varshney, M. J. Najmussehar, and M. J. Ahsan, "Estimation of more than one parameters in stratified sampling with fixed

budget," *Mathematical Methods of Operations Research*, vol. 75, no. 2, pp. 185–197, 2012.

[43] E. M. L. Beale, "On minimizing a convex function subject to linear inequalities," *Journal of the Royal Statistical Society: Series B*, vol. 17, no. 2, pp. 173–184, 1955.

[44] L. Kish, "Optima and proxima in linear sample designs," *Journal of the Royal Statistical Society: Series A*, vol. 139, no. 1, pp. 80–95, 1976.

[45] K. G. Reddy and M. G. M. KhanKhan, "stratifyR: an R Package for optimal stratification and sample allocation for univariate populations," *Australian & New Zealand Journal of Statistics*, vol. 62, no. 3, pp. 383–405, 2020.

[46] Lingo, "LINGO-User's Guide," North Dayton Street, Chicago, Illinois, 60622, USA, 2016.

Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 6602155, 14 pages
https://doi.org/10.1155/2022/6602155

Research Article

Fixture Design in Flexible Tooling of Aircraft Panel Based on Thin Plate Theory

Zemin Pan,[1] **Ying Liu**,[2] **Zhichao Sun**,[3] **Songyang Chang**,[3] **and Qiang Fang**[4]

[1]*School of Information Science and Engineering, NingboTech University, 1 South Qianhu Rd, Ningbo 315100, China*
[2]*AVIC Xian Aircraft Industry Group Company LTD, 1 Xifei Rd, Yanliang Dist, Xian 710089, China*
[3]*School of Mechanical Engineering, Zhejiang University, 38 Zheda Rd, Hangzhou 310027, China*
[4]*State Key Lab of Fluid Power Transmission and Control, Zhejiang University, 38 Zheda Rd., Hangzhou 310027, China*

Correspondence should be addressed to Zemin Pan; zeminpan@zju.edu.cn

Received 10 December 2021; Revised 2 March 2022; Accepted 4 March 2022; Published 25 April 2022

Academic Editor: Guoqiang Wang

Flexible and conformal positioning of the skin is of great significance for the efficient and low deformation assembly of aircraft panels. As the main conformal positioning part of the skin, the fixture positioning surface is complicated to design when taking into account the flexible positioning. In this study, taking the low deformation and flexible clamping of the skin under complex stress conditions as the target, a fixture positioning profile design method is proposed to predict the deformation of the skin and provide moderate support for the skin based on thin plate theory (TPT). In the face of complex stress and boundary conditions of the skin, the differential method is used to solve the stress and deformation of the skin under different support conditions, which is correctly verified by finite element simulation. The proposed method reveals the influence of the fixture positioning profile on the deflection of the aircraft skin under concentrated load (drilling force), which provides a good design theoretical basis for the fixture design for the low deformation flexible clamping of the skin.

1. Introduction

In the fuselage, wing, and other structures of aircraft, thin-walled parts are widely used to reduce the weight of the whole aircraft. The clamping deformation and assembly deviation of this kind of thin-walled parts in the assembly stage will accumulate and spread along the assembly dimension chain of products [1] and ultimately affect the quality of aircraft manufacturing [2]. So how to design a suitable fixture for thin-walled parts is of great significance in the field of aircraft assembly.

Aiming at the problem of fixture design for weak stiffness parts, Cai et al. [3] came up with the "N-2-1" positioning principle in 1996. The "N-2-1" positioning principle is more suitable for the clamping of thin-walled parts than the "3-2-1" positioning principle by increasing the number of positioning elements on the main positioning surface and enhancing the overall stiffness of thin-walled parts. The increase in the number of positioning elements causes trouble in the fixture arrangement. The intelligent optimization algorithm has the characteristics of high search efficiency and convenient mathematical description, which is commonly used in fixture positioning point layout optimization. For fixture layout optimization, different fixtures develop different solutions. Minh et al. [4] presented a single-axis rod type flexible fixture system for thin-walled components in machining processes utilizing the N-2-1 location principle and proposed a geometry-based method to find the optimal location of the workpiece relative to the fixture system, which could find the optimal location within 30 s. Chen et al. [5] proposed a "N-M" positioning principle based on the "N-2-1" principle and used the genetic algorithm to search the optimal solution of fixture layout, which reduced the machining deformation of thin-walled parts by 56.5%. Abolfazl et al. [6] used the genetic algorithm to optimize the unconstrained problem of car body parts

clamping error propagation and realized the optimal fixture configuration. Ma et al. [7] proposed a hybrid optimization algorithm GAOT, which generated the global optimal fixture layout, and obtained a better optimal solution than the existing optimization algorithm. Zhou et al. [8] designed the fixture layout scheme based on the hybrid particle swarm optimization algorithm with the objective of minimizing the normal deformation of metal sheet in the flexible tooling system. Wan et al. [9] applied the fast nondominated multiobjective optimization algorithm (NSGA-II) to optimize the four defined objective functions to minimize the position error of the workpiece and obtain the optimal positioner layout. Yang et al. [10] proposed a fixture location layout optimization method based on the support vector regression model and NSGA-II algorithm and realized the optimal design of support points for aircraft skin parts. Zeshan et al. [11] proposed a "N-3-2-1" positioning principle for sheet metal fixture layout optimization and combined with finite element and genetic algorithm to realize fixture layout optimization in the multipoint spot welding process. Li et al. [12] proposed a method of fixture location layout optimization design for thin-walled parts based on the Kriging agent model and flower pollination algorithm and completed the "4-2-1" fixture layout optimization design for curved thin-walled parts. Aimed at aircraft weak stiffness parts, Wang et al. [13] calculated the maximum deformation under different positioning schemes by finite element simulation and combined with the firefly algorithm to achieve the iterative optimization of positioning point layout. Li et al. [14] proposed an optimization method for the clamping scheme of aircraft thin-walled parts based on the genetic algorithm. Combined with finite element simulation, the synchronous optimization of clamping layout and clamping sequence was realized. To sum up, in the design of thin-walled parts fixture, the application of "N-2-1" positioning principle, and the positioning point layout optimization guided by the intelligent optimization algorithm, can significantly reduce the deformation of thin-walled parts in the assembly and manufacturing process and improve the final quality of products.

In the processing of thin-walled parts, the workpiece and cutting tool will have elastic deformation under the action of cutting force, which will lead to the change of process parameters and affect the processing quality. Thin plate problem is a basic problem in elastic mechanics. The key to study this kind of problem is how to deal with the boundary conditions and how to get the exact analytical or numerical solution [15–17]. Based on the thin plate theory of elasticity, the machining deformation of thin-walled parts can be analyzed. The solution of thin plate problems usually includes the finite element method, meshless method, finite difference method, boundary element method, Rayleigh–Ritz method, and Galerkin method [18]. Slark et al. [19] studied the large deformation problem of simply supported thin plates by the meshless method and solved the nonlinear equations. Based on Kirchhoff plate theory, Shabana et al. [20] proposed a 12-node plate element model, in which the displacement continuity can be ensured by rigid connection between nodes, and the motion and deformation of flexible plate can be accurately described. Chen et al. [21] proposed a method of machining error compensation for thin-walled parts based on bicubic B-spline interpolation considering the influence of elastic deformation. In order to reduce the machining deformation of thin-walled titanium alloy parts, Li et al. [22] proposed a nonuniform allowance design strategy of discrete allowance volume element based on the Rayleigh Ritz method, which makes better use of the workpiece stiffness and reduces the machining error. Tang et al. [23] solved the bending problems of Kirchhoff and Winkler thin plates by using the generalized finite difference method. Numerical experiments show that the method can effectively solve the bending problems of two kinds of thin plates under different transverse loads.

In addition, as a large aviation thin-walled part, the assembly sequence of aircraft panels has a crucial impact on the assembly efficiency and assembly quality of the panels. In order to improve assembly efficiency and quality, many scholars home and abroad began to study the mechanical assembly sequence as early as 4 decades ago. The optimization of the assembly sequence in the early stage was mainly realized by manual calculation, and its defects were low efficiency and single results. With the rapid development of computer computing power and the introduction of intelligent algorithms, the optimization of assembly sequence has become efficient, flexible, and rich in calculation results [24]. Yang et al. [25] developed a novel PSOBC algorithm based on the conventional PSO algorithm for complex products; this method avoids a large number of matrix calculation by establishing synchronized assembly Petri net (SAPN) to describe the precedence relationships, and the provided PSOBC not only prevents premature convergence to a high degree but also keeps a more rapid convergence rate than the standard PSO algorithm. Bahubalendruni et al. [26] proposed a novel and efficient method to obtain all valid assembly sequences and optimized assembly sequence for a given assembled product, which considers four basic predicates, namely, "liaison predicate, geometrical feasibility, mechanical feasibility, and stability" to validate each sequence, and the validity of the method is validated by different example products. Bahubalendruni et al. [27] proposed an efficient computational method to find the optimal sequence from a huge set of all assembly sequences, and the method is proven in generating optimal solutions for any given product effectively.

Based on the above references, it can be found that both scholars domestic and abroad have made a lot of achievements in the research of thin-walled parts fixture design. As a kind of large thin-walled parts, how to reduce the load deformation of aircraft skin and achieve reliable clamping is the paramount problem of aircraft panel flexible tooling development. Scholars domestic and abroad have made research on the fixture layout and deformation analysis of thin-walled parts, but the research on the fixture layout mainly focuses on the point supporting flexible tooling, and the research on the typical fixture type tooling like surface supporting is too little. Few scholars established the mechanical model to guide fixture design through elastic deformation theory of aircraft skins. In this study, in order to

minimize the deformation of aircraft skins under load, the research on the design of flexible fixture is carried out.

2. Analysis of Fixture Design

The skin of large aircraft is usually designed with variable thickness. The thickness of the skin is reduced in the area with less stress, which forms the inner surface of the skin as concave, so as to reduce the weight of the whole skin structure. Figure 1 shows the skin of the front fuselage panel of a certain civil aircraft, and the blue areas are the chemical milling area of the skin. When it comes to the flexible tooling fixture, the chemical milling area of different skin may be different, which makes the fixture surface difficult to coordinate or even share the same fixture surface. In an ideal situation, the longer the length of the fixture surface is, the longer the length of the effective bonding section with the skin is, and the better the fixture stiffness can be achieved. However, due to the location of the truss axis, the length of the clamping surface is limited within a certain range, as shown in Figure 2 Therefore, it is necessary to study the quantitative relationship between the length of the surface and the effect of skin support stiffness to guide the design of the surface.

In this study, aiming at the design problem of the fixture, the design goal is to ensure the effect of skin support, and the solution of thin plate problem in the elasticity is used to guide the fixture surface design according to the suppression effect of the elastic deformation of the skin.

3. Mechanical Model Based on Thin Plate Theory

In the assembly process, the skin is affected by gravity, strap pressure, board support force, frictional force, and drilling force, so it is often necessary to establish a complex mechanical analysis model, which increase the difficulty of solving, so the establishment of a simplified mechanical model is an important part of solving engineering problems. For the case where the skin is thin and large and only the stress-strain conditions near small load area are analyzed, the classical thin plate deformation theory is adopted, where the skin radian is ignored, the skin is considered to be an ideal elastic body, and the theoretical model is based on elasticity of a rectangular thin plate. Hence, several assumptions should be given in advance [28]:

(1) Zero radian assumption: since the stress area is small when drilling, the surface radian of the skin can be ignored;

(2) Continuity assumption: it is assumed that the object is a continuous medium and there is no gap between particles;

(3) Complete elasticity assumption: the object completely obeys Hooke's law, and the strain of the object is proportional to the stress;

(4) Isotropic assumption: the physical properties of any point in the object are the same in all directions;

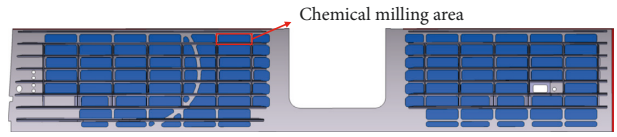


Figure 1: Schematic diagram of skin chemical milling area.

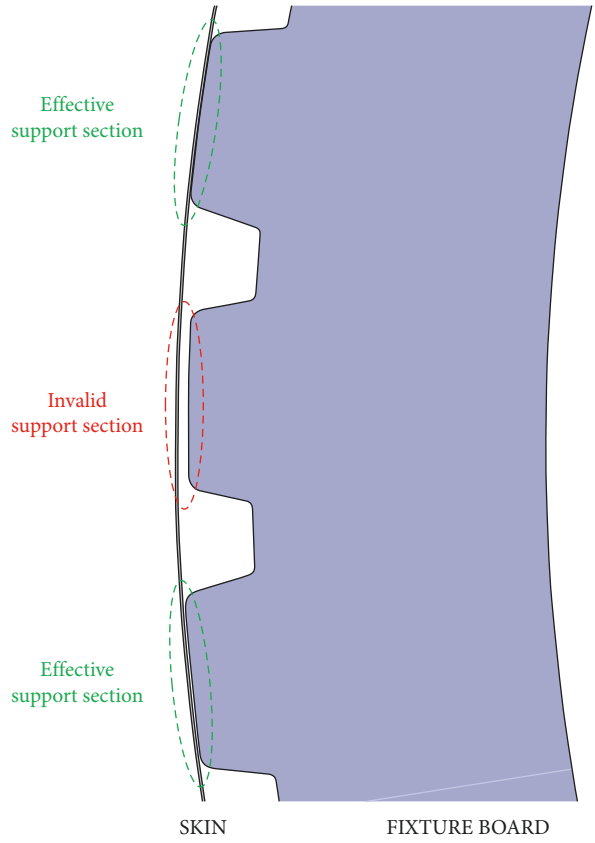


Figure 2: Schematic diagram of fixture with skin fitting.

(5) Uniformity assumption: the object is composed of the same type of uniform material, so that the elastic coefficient of the object does not change with the change of position coordinates.

The thin plate will be bent under the action of transverse load, and the stress, strain, and displacement problems caused by it are calculated according to the bending problem of thin plate [29]. The bending problems of thin plates are generally divided into small deflection bending problems and large deflection bending problems. After the skin is positioned and clamped on the frame, the preconnection holes between the skin and truss, and the skin and corner piece need to be drilled. In this process, the fixed effect of the strap and the clamp on the skin is regarded as the fixed boundary condition of the skin, and the skin is only affected by the drilling force. The drilling force is a transverse force and perpendicular to the neutral plane. Therefore, the stress and deformation of the skin under the action of drilling force can be solved as the bending problem of thin plate.

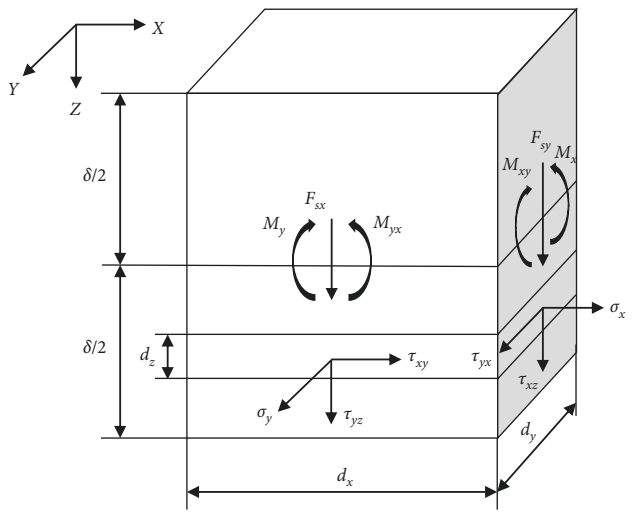


Figure 3: Stress analysis of thin plate microunit.

When dealing with the problem of thin plate bending, it can be considered as the problem of small deflection bending of thin plate when the stiffness of thin plate is large and the deflection is far less than the thickness under the action of external load. The Kirchhoff–Love theory [29] should be satisfied.

(1) On any normal line perpendicular to the neutral plane, the displacement of each point in the thickness of the thin plate is the same, that is, the deflection ω, which is only related to the x and y coordinates. Let the neutral plane of the thin plate be the xy plane (Figure 3), so that

$$w = w(x, y). \tag{1}$$

(2) The deformation caused by the normal stress σ_z is negligible, so the equation can be simplified to

$$\left.\begin{aligned} e_x &= \frac{1}{E}\left(\sigma_x - \nu\sigma_y\right) \\ e_y &= \frac{1}{E}\left(\sigma_y - \nu\sigma_x\right) \\ \gamma_{xy} &= \frac{2(1+\nu)}{E}T_{xy} \end{aligned}\right\}. \tag{2}$$

(3) There is only vertical displacement and no displacement parallel to the neutral plane in the thin plate, that means

$$(\mu)_{z=0} = (v)_{z=0} = 0. \tag{3}$$

The bending deformation problem of the skin is simplified to the small deflection bending problem of the thin plate, and a differential equation for $\omega(x, y)$ needs to be established. As shown in Figure 3, take out a parallelepiped microunit, whose length and width are dx and dy, respectively, from the rectangular thin plate with thickness δ. The four sides of the unit are subjected to the internal force of the plate, the upper surface is subjected to the external load $qdxdy$, and the lower surface has no load. On the side where x is a constant, stress components s_x, τ_{xy}, and τ_{xz} act; on the side where y is a constant, stress components σy, τ_{yx} (equal to τ_{xy}), and τ_{yz} act. These stresses can be synthesized into bending moments M_x, M_y, torque M_{xy}, and transverse shear forces F_{sx}, F_{sy}, respectively.

Combine the stress components per unit width of the flank:

$$\left.\begin{aligned} M_x &= \int_{-\frac{\delta}{2}}^{\frac{\delta}{2}} \sigma_x z\, dz\,, \\ M_y &= \int_{-\frac{\delta}{2}}^{\frac{\delta}{2}} \sigma_y z\, dz\,, \\ M_{xy} &= M_{yx}\int_{-\frac{\delta}{2}}^{\frac{\delta}{2}} \tau_{xy} z\, dz\,, \end{aligned}\right\} \tag{4}$$

Combining (2) and elastic mechanics, it can be known that the generalized Hooke's law expression for deflection $\omega(x, y)$ is

$$\left.\begin{aligned} \sigma_x &= -\frac{E_z}{1-\nu^2}\left(\frac{\partial^2 w}{\partial x^2} + \nu\frac{\partial^2 w}{\partial y^2}\right), \\ \sigma_y &= -\frac{E_z}{1-\nu^2}\left(\frac{\partial^2 w}{\partial y^2} + \nu\frac{\partial^2 w}{\partial x^2}\right), \\ \tau_{xy} &= -\frac{E_z}{1-\nu^2}\frac{\partial^2 w}{\partial x\,\partial y}. \end{aligned}\right\} \tag{5}$$

Introducing the bending stiffness of the thin plate (the dimension is L^2MT^{-2}),

$$D = \frac{E\delta^3}{12\left(1-\nu^2\right)} \tag{6}$$

Substituting (5) into (4), the bending moment and torque of the microunit can be obtained as

$$\left.\begin{aligned} M_x &= D-\left(\frac{\partial^2 w}{\partial x^2}+\nu\frac{\partial^2 w}{\partial y^2}\right),\\ M_y &= D-\left(\frac{\partial^2 w}{\partial y^2}+\nu\frac{\partial^2 w}{\partial x^2}\right),\\ M_{xy} &= M_{yx} = -D(1-\nu)-\frac{\partial^2 w}{\partial x\,\partial y}. \end{aligned}\right\} \tag{7}$$

In order to keep balance of the microunit, the sum of the moments around x-axis and y-axis is 0, and the force in z-direction is balanced, that means the following balance conditions must be satisfied:

$$\begin{aligned} \sum M_x &= 0,\\ \sum M_y &= 0,\\ \sum F_z &= 0. \end{aligned} \tag{8}$$

After omitting the high-order trace, it can be obtained that

$$\left.\begin{aligned} \frac{\partial M_x}{\partial x}+\frac{\partial M_{yx}}{\partial y} &= F_{sx}\\ \frac{\partial M_y}{\partial y}+\frac{\partial M_{xy}}{\partial x} &= F_{sy} \end{aligned}\right\}, \tag{9}$$

and

$$\frac{\partial F_{sx}}{\partial x}+\frac{\partial F_{sy}}{\partial y}+q=0. \tag{10}$$

Regarding Mxy=Myx, substitute (7) and (9) into equation (10) to get

$$\frac{\partial^4 w}{\partial x^4}+2\frac{\partial^4 w}{\partial x^2\partial y^2}+\frac{\partial^4 w}{\partial y^4}-\frac{q}{D}=0. \tag{11}$$

Arrange (11) as follows, which is the equilibrium differential equation for bending deformation of the skin with small deflection.

$$\nabla^4 w=\frac{q}{D}, \tag{12}$$

where $\nabla^2=\partial^2/\partial x^2+\partial^2/\partial y^2$ is the Laplace operator.

However, the assumption for small deflection theory is that the longitudinal displacement of thin plate, which is perpendicular to the neutral plane is so small than the thickness of thin plate that the membrane force can be ignored. For a thin metal plate such as the aircraft skin, the longitudinal displacement of each point in the plane is not necessarily much less than the thickness of the plate. If so, it can be solved as a large deflection bending problem of thin plate. When establishing the equilibrium differential equation of large deflection bending problem of thin plate, it is necessary to take into account the neutral plane strain and membrane force caused by the longitudinal displacement of each point in the neutral plane.

The bending differential equation of the elastic plate is

$$D\nabla^4 w=\left(F_{tx}\frac{\partial^2 w}{\partial x^2}+F_{ty}\frac{\partial^2 w}{\partial y^2}+2F_{txy}\frac{\partial^2 w}{\partial x\,\partial x}\right)+q. \tag{13}$$

The deflection ω and the film forces (caused by lateral load q) F_{tx}, F_{ty}, and F_{txy} in this equation are unknown. In order to simplify the equation and reduce the number of unknowns, the stress function Φ is introduced, so that

$$\left.\begin{aligned} F_{tx} &= \delta\sigma_x=\delta\frac{\partial^2\phi}{\partial y^2},\\ F_{ty} &= \delta\sigma_y=\delta\frac{\partial^2\phi}{\partial x^2},\\ F_{txy} &= \delta\tau_{xy}=-\delta\frac{\partial^2\phi}{\partial x\,\partial y}. \end{aligned}\right\} \tag{14}$$

At this point, the elastic plate differential equation becomes an equation containing two unknown ω and Φ, which still cannot be solved. A compatibility equation needs to be constructed for the relationship between film force and deflection. Considering the geometric relationship between the normal strain, shear strain, and three-dimensional displacement in the microunit of the thin plate, the following geometric relationship equation can be obtained:

$$\left.\begin{aligned} \varepsilon_x &= \frac{\partial u}{\partial x}+\frac{1}{2}\left(\frac{\partial w}{\partial x}\right)^2,\\ \varepsilon_x &= \frac{\partial u}{\partial y}+\frac{1}{2}\left(\frac{\partial w}{\partial y}\right)^2,\\ \gamma_{xy} &= -\frac{\partial u}{\partial x}+\frac{\partial u}{\partial x}+\frac{\partial w}{\partial x}\frac{\partial w}{\partial y}. \end{aligned}\right\} \tag{15}$$

By eliminating the lateral displacements u and v in the equation, the following compatibility equation can be obtained:

$$\left(\frac{\partial^2 w}{\partial x\,\partial y}\right)^2-\frac{\partial^2 w}{\partial x^2}\frac{\partial^2 w}{\partial y^2}=\frac{\partial^2\varepsilon_y}{\partial x^2}+\frac{\partial^2\varepsilon_x}{\partial y^2}-\frac{\partial^2\gamma_{xy}}{\partial x\,\partial y}. \tag{16}$$

According to (2) and (14), the compatibility equation can be simplified; then, the differential equations of elastic plate can be simultaneously combined to obtain the following differential equation system based on the large deflection theory:

$$\left.\begin{aligned} \nabla^4 w &= \frac{q}{D}+\frac{\delta}{D}\left(\frac{\partial^2\phi}{\partial x^2}\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2\phi}{\partial y^2}\frac{\partial^2 w}{\partial x^2}-2\frac{\partial^2\phi}{\partial x\,\partial y}\frac{\partial^2 w}{\partial x\,\partial y}\right),\\ \nabla^4\phi &= E\left[\left(\frac{\partial^2 w}{\partial x\,\partial y}\right)^2-\frac{\partial^2 w}{\partial x^2}\frac{\partial^2 w}{\partial y^2}\right]. \end{aligned}\right\} \tag{17}$$

4. Theoretical Calculation Based on the Difference Method

In the process of using small deflection theory or large deflection theory to solve the skin deformation problem, the same boundary of the skin may contain both fixed boundary and free boundary or other types of boundaries due to the panel is partiality supported by the board. Moreover, during the drilling or milling, the distribution of the load and size of the load is irregular; thus, the description of the load is also complex. To sum up, there are mathematical difficulties in solving the differential equation of skin deformation because the mathematical models are often not that accurate to include the different boundary conditions of different skin segments and the randomly distributed load conditions.

Difference method is a common method to solve partial differential equations. The basic method is as follows: the domain is discretized by difference grid, the independent variable is represented by the function of the discrete variable at the point of difference grid, and the derivative or partial derivative is represented by the difference quotient of the function between the points of difference grid. The original differential equation and boundary condition are approximately expressed by difference equation in the form of algebraic equation, and then, the differential equation is replaced by the algebraic equation. It is considered that the solution of the algebraic equation is the approximate solution of the original problem.

4.1. Difference Calculation Based on Small Deflection Theory. For the convenience of calculation, the skin area is divided by square mesh, and the side length of square mesh is represented by Δd. As shown in Figure 4, the skin is divided into $(i-1)\times(j-1)$ mesh regions, including $i \times j$ nodes.

Using the square grid as shown in Figure 4, from column x = 1 to column x = i, write the difference equations for the unknown nodes in each column and list the difference equations. By using the matrix form, we can get the following results:

$$KW = Q, \tag{18}$$

where K is the coefficient matrix of order $(i\times j)\times(i\times j)$ of the difference equations, which can be expressed as

$$K = \begin{bmatrix}
C_1 & B_1 & A_1 & & & & & & & \\
B_2 & C_2 & D_2 & A_2 & & & & & & \\
A & B & C & B & A & & & & & \\
 & A & B & C & B & A & & & & \\
 & & & \ddots & & & & & & \\
 & & & A & B & C & B & A & & \\
 & & & & A & B & C & B & A & \\
 & & & & & A_2' & D_2' & C_2' & B_2' & \\
 & & & & & & A_1' & B_1' & C_1' &
\end{bmatrix}_{(I\times J)\times(I\times J)}, \tag{19}$$

where $A, B, C, A_1, B_1, C_1, A_2, B_2, C_2, D_2, A_1', B_1', C_1', A_2', B_2', C_2', D_2'$ are the block matrices, all of which are square matrices of order $j \times j$. If the number of rows is j and the number of columns is i in this matrix, A, B, and C are the coefficient matrices corresponding to the internal i-4 column nodes, $A_2, B_2, C_2, D_2, A_1', B_1', C_1', A_2', B_2', C_2', D_2'$ are the coefficient matrices corresponding to a column of nodes in the boundary, and A_3, B_3, C_3, A_3', B_3', and C_3' are the coefficient matrices corresponding to the nodes of the boundary column. In the construction, the above matrix needs to be adjusted according to the actual boundary conditions.

W is a $(i \times j)\times 1$-order column matrix composed of unknown deflection $w_{i,j}$ at each node and is expressed as follows:

$$W = \begin{bmatrix} W_1 \\ W_2 \\ W_3 \\ \vdots \\ W_i \end{bmatrix}_{(i\times j)\times 1}, \qquad W_i = \begin{bmatrix} w_{1,1} \\ w_{1,2} \\ w_{1,3} \\ \vdots \\ w_{1,j} \end{bmatrix}_{(j\times 1)}. \tag{20}$$

Q is the value on the right side of the difference equation and is a known condition in actual calculation, which can be expressed as

$$Q = \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ \vdots \\ Q_i \end{bmatrix}_{(i\times j)\times 1}, \qquad Q_i = \begin{bmatrix} Q_{i,1} \\ Q_{i,2} \\ Q_{i,3} \\ \vdots \\ Q_{i,j} \end{bmatrix}_{j\times 1}, \tag{21}$$

$$q_{i,j} = \frac{q(i,j)}{D}\Delta d^4,$$

where $q(i, j)$ is the load value per unit area at the node (i,j).

When the boundary condition is free, the virtual nodes of one line outside the boundary can be represented by two lines of nodes close to the boundary and the virtual nodes of two lines outside the boundary can be represented by three lines of nodes close to the boundary. Therefore, when constructing the coefficient matrices A, B, and C corresponding to the internal i-4 column nodes, the j-4 between them acts as the conventional coefficient, which is derived from the original fourth-order difference equation. The first two lines and the second two lines of coefficient matrices A, B, and C are still missing the virtual nodes of the line outside the boundary and the two lines of virtual nodes outside the boundary. Therefore, it is necessary to rewrite the correlation coefficient in the matrix, as shown in t_1–t_9 and t_1'–t_9' in the following formula. Since the coefficients to be rewritten in dealing with simply supported boundary and fixed boundary have been included in t_1–t_9 and t_1'–t_9', the coefficient matrices A, B, and C can be expressed in the following general form:

$$A = \begin{bmatrix} t_1 & & & & & & \\ & 1 & & & & & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & & & 1 & & \\ & & & & & 1 & \\ & & & & & & t_1' \end{bmatrix},$$

$$B = \begin{bmatrix} t_2 & t_3 & & & & & \\ t_4 & -8 & 2 & & & & \\ & 2 & -8 & 2 & & & \\ & & & \ddots & & & \\ & & & 2 & -8 & 2 & \\ & & & & 2 & -8 & t_4' \\ & & & & & t_3' & t_2' \end{bmatrix}, \quad (22)$$

$$c = \begin{bmatrix} t_5 & t_6 & t_7 & & & & \\ t_8 & t_9 & -8 & 1 & & & \\ 1 & -8 & 20 & -8 & 1 & & \\ & & & \ddots & & & \\ & & 1 & -8 & 20 & -8 & 1 \\ & & & 1 & -8 & t_9' & t_8' \\ & & & & t_7' & t_6' & t_5' \end{bmatrix}.$$

After the construction of coefficient matrix K and matrix Q, the numerical solution of skin deflection under different boundary conditions can be obtained by using the MATLAB software, and then, the real solution of the original small deflection bending differential equation can be approximated. We can use the left division command in MATLAB, which has better numerical stability than calculating the inverse matrix [30]. The calculation formula is as follows:

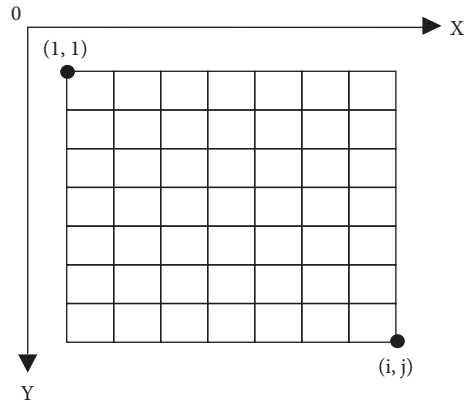


Figure 4: Differential mesh generation.

$$W = \frac{K}{Q}. \quad (23)$$

4.2. Difference Calculation Based on Large Deflection Theory. According to (17), the difference equations based on large deflection theory are composed of differential equations and compatibility equations of elastic surface. To solve this problem, we need to solve the unknown functions in the equations, deflection function $W(x, y)$ and stress function $\phi(x, y)$, according to the actual boundary condition of the mechanical model. Since the exact solutions of multiple nonlinear differential equations is difficult to obtain, based on the difference calculation method, the equilibrium differential equations of large deflection theory are transformed into the following difference equations:

$$\left.\begin{aligned} K_1 W = Q, \\ K_2 \phi = G. \end{aligned}\right\} \quad (24)$$

There are two unknown matrices in the difference equations, which are deflection matrix W and stress function matrix ϕ. In the original system of differential equations, the unknown matrix ϕ is on the right side of the differential equation of elastic surface. In order to unify the coefficients, it is moved to the left side, that is, the unknown matrix K_1 is included in ϕ. G is a matrix containing an unknown matrix W. K_1 and K_2 are the coefficient matrices to be constructed. The equations cannot be solved directly by the difference method. The approximate solution is obtained by using the idea of gradual approximation. First, the initial value of ϕ is assumed and brought into the differential equation of elastic surface to solve W by the difference method (usually, the initial value of ϕ is 0, which is solved according to the small deflection theory). Then, the solved W is brought into the compatible equation to solve the ϕ value by the difference method, and then, ϕ is brought into the differential equation of elastic surface to compare the solved W with the previous W. It is considered that the solution is completed by iterating in cycles according to this law until the W values of the previous and the following two times approach.

The approximation conditions of the two W values (W^t and W^{t+1}) are expressed by the Frobenius norm of the difference between the two matrices:

$$\left\|W^t - W^{t+1}\right\|_F = \sqrt{\sum_{m=1}^{i}\sum_{n=1}^{i}\left|w_{m,n}^t - w_{m,n}^{t+1}\right|^2} \le 0.01. \quad (25)$$

The differential equation of elastic surface is solved. According to the difference idea, the difference relation of each basic node $w_{i,j}$ is represented by the difference diagram as shown in Figure 5.

It can be seen that the partial differential expression of the stress function $\phi(x, y)$ is needed when constructing the coefficient matrix of the difference equations. The second-order partial differential of stress function $\phi(x, y)$ can be expressed by calculating the second-order degree of matrix ϕ twice with gradient function ϕ in MATLAB software. At the same time, it should be noted that in the difference equation of the basic node $\omega_{i,j}$, the partial differential of ϕ in its coefficient must be the second step degree of one-to-one corresponding node $\phi_{i,j}$ in the stress function matrix ϕ. In the coefficient matrix, the coefficients $\partial^2\phi/\partial x^2$, $\partial^2\phi/\partial y^2$, and $\partial^2\phi/\partial x\,\partial y$ used by $\omega_{i,j}$ in the corresponding rows of the coefficient matrix are the two-step values of $\phi_{i,j}$ in x, y, and xy directions, respectively. Therefore, the difference between the large deflection theory and the small deflection theory is that when the large deflection theory is used, the values of each line in the coefficient matrix are affected by ϕ and change with ϕ. In the process of solving the differential equation of elastic surface in large deflection theory, the treatment of boundary conditions is the same as that in solving the differential equation of small deflection theory. The coefficient matrix K_j of elastic surface differential equation can be expressed as follows:

$$K_1 = \begin{bmatrix} C_1 & D_1 & H_1 & & & & & & \\ B_2 & C_2 & B_D & H_2 & & & & & \\ A_3 & B_3 & C_3 & D_3 & H_3 & & & & \\ & A_4 & B_4 & C_4 & D_4 & H_4 & & & \\ & & & \ddots & & & & & \\ & & & A_{i-3} & B_{i-3} & C_{i-3} & D_{i-3} & H_{i-3} & \\ & & & & A_{i-2} & B_{i-2} & C_{i-2} & D_{i-2} & H_{i-2} \\ & & & & & A_{i-1} & B_{i-1} & C_{i-1} & D_{i-1} \\ & & & & & & A_i & B_i & C_i \end{bmatrix}_{(i\times j)\times(i\times j)}. \quad (26)$$

In the matrix, A_i, B_i, C_i, D_i, and H_i are the square matrices of order j, which can be given by the following equation (27).

$$A_i = \begin{bmatrix} \left(t_i^a, 1\right)_b & & & & & & \\ & 1 & & & & & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & & & 1 & & \\ & & & & & 1 & \\ & & & & & & \left(t_i^a, 1\right)_b \end{bmatrix},$$

$$B_i = \begin{bmatrix} \left(t_{i,1}^d\right)_b & \left(t_{i,1}^e\right)_b & & & & & \\ \left(t_{i,2}^c\right)_b & t_{i,2}^d & t_{i,2}^e & & & & \\ & t_{i,3}^c & t_{i,3}^d & t_{i,3}^e & & & \\ & & & \ddots & & & \\ & & & t_{i,j-2}^c & t_{i,j-2}^d & t_{i,j-2}^e & \\ & & & & t_{i,j-1}^c & t_{i,j-1}^c & \left(t_{i,j-1}^c\right)_b \\ & & & & & \left(t_{i,j}^c\right)_b & \left(t_{i,j}^d\right)_b \end{bmatrix},$$

$$C_i = \begin{bmatrix} \left(t_{i,1}^f\right)_b & \left(t_{i,1}^g\right)_b & \left(t_{i,1}^h\right)_b & & & & \\ \left(t_{i,2}^g\right)_b & \left(t_{i,2}^f\right)_b & t_{i,2}^g & 1 & & & \\ 1 & t_{i,3}^g & t_{i,3}^f & t_{i,3}^g & 1 & & \\ & & & \ddots & & & \\ & & 1 & t_{i,j-2}^g & t_{i,j-2}^f & t_{i,j-2}^g & 1 \\ & & & 1 & t_{i,j-1}^g & \left(t_{i,j-1}^f\right)_b & \left(t_{i,j-1}^f\right)_b \\ & & & & \left(t_{i,j}^h\right)_b & \left(t_{i,j}^g\right)_b & \left(t_{i,j}^f\right)_b \end{bmatrix},$$

$$D_i = \begin{bmatrix} \left(t_{i,1}^d\right)_b & \left(t_{i,1}^c\right)_b & & & & & \\ \left(t_{i,2}^e\right)_b & t_{i,2}^d & t_{i,2}^c & & & & \\ & t_{i,3}^e & t_{i,3}^d & t_{i,3}^c & & & \\ & & & \ddots & & & \\ & & & t_{i,j-2}^e & t_{i,j-2}^d & t_{i,j-2}^c & \\ & & & & t_{i,j-1}^e & t_{i,j-1}^d & \left(t_{i,j-1}^c\right)_b \\ & & & & & \left(t_{i,j}^e\right)_b & \left(t_{i,j}^d\right)_b \end{bmatrix},$$

$$H_i = \begin{bmatrix} \left(t_i^b, 1\right)_b & & & & & & \\ & 1 & & & & & \\ & & 1 & & & & \\ & & & \ddots & & & \\ & & & & 1 & & \\ & & & & & 1 & \\ & & & & & & \left(t_i^b, 1\right)_b \end{bmatrix}. \quad (27)$$

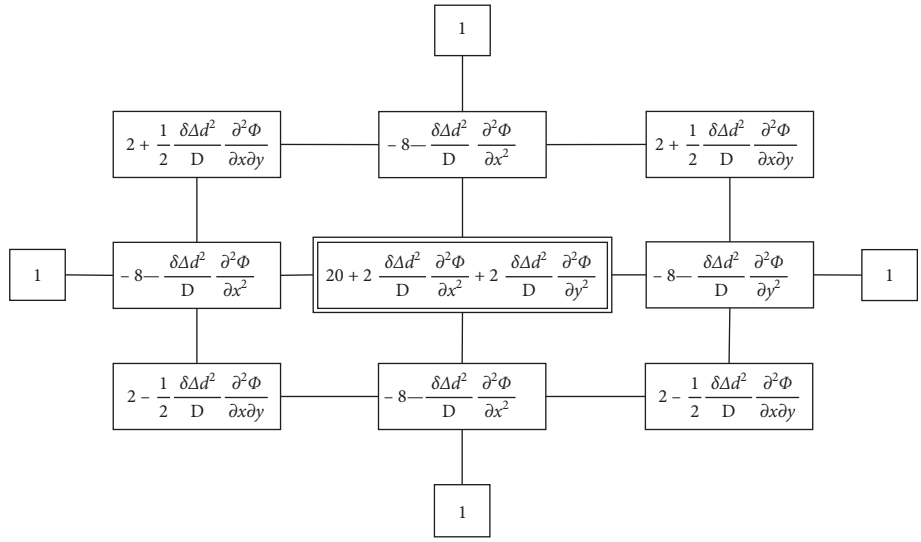


Figure 5: Difference diagram of nodal relation in large deflection theory.

Similar to the boundary treatment method based on the theory of small deflection, the corresponding coefficients of the first two and the latter two lines of A_i, B_i , C_i, D_i, and H_i should be adjusted according to the change of boundary conditions, and the parameters with b subscripts are used in the matrix. Other parameters are given in Table 1.

The compatibility equation is solved. The boundary condition of stress function ϕ is treated as follows: the ϕ value of the boundary node is 0, and the ϕ value of the virtual node of a row (or column) outside the boundary is the ϕ value of the symmetric node of a row (or column) inside the boundary. The construction form of coefficient matrix can refer to the construction method of coefficient matrix in the calculation of small deflection theory. The right term of the compatibility equation also deals with the deflection matrix W by using gradient function.

5. Calculation Error Analysis

In this section, the "4-2-1" support layout of skin is taken as an example to evaluate the accuracy of the elastic mechanics model by using the difference calculation method based on the small deflection theory and the one based on the large deflection theory.

Taking the thickness of the skin as 2 mm, two fixtures are used, and each fixture has two sections of surface to fit with the skin. The concentrated force is applied on the middle part of the skin to calculate its deflection. As shown in Figure 6, the loads and constraints of the skin are shown, AB and CD are the free boundaries and AD and BC are the free-fixed composite boundaries. In the case of free-fixed composite boundary conditions, it is difficult to solve the analytical solution of the differential equation of thin plate bending deformation by using conventional methods. Therefore, using the difference method to deal with the complex boundary problems can fully reflect the convenience of the difference method to deal with the complex boundary conditions. In order to evaluate the accuracy of the

Table 1: Corresponding parameter values of coefficient matrix of large deflection theory.

Parameter	Corresponding value
$t^e_{i,j}$	$2 + 1/2\partial\Delta\ d^2/D\partial^2\phi_{i,j}/\partial x\ \partial y$
$t^d_{i,j}$	$-8 - 1/2\partial\Delta\ d^2/D\partial^2\phi_{i,j}/\partial\partial y^2$
$t^e_{i,j}$	$2 - 1/2\partial\Delta\ d^2/D\partial^2\phi_{i,j}/\partial x\ \partial y$
$t^f_{i,j}$	$20 + 2\partial\Delta\ d^2/D\partial^2\phi_{i,j}/\partial x^2 + 2\partial\Delta\ d^2/D\partial^2\phi_{i,j}/\partial y^2$
$t^g_{i,j}$	$-8 - \partial\Delta\ d^2/D\partial^2\phi_{i,j}/\partial x^2$

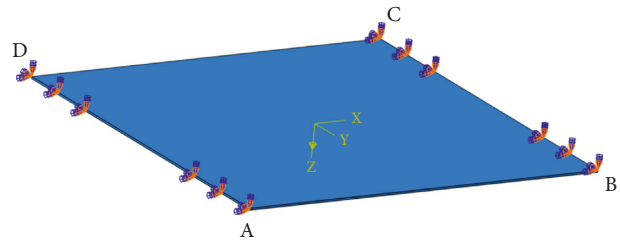


Figure 6: Skin loads and constraints.

theoretical calculation, ABAQUS finite element simulation is used for comparative calculation. Aluminum alloy 2024-T3 and SC8R shell elements are selected as the skin material in the simulation.

First, the deformations of two kinds of the skin with dimensions of 200×200 mm, fixed boundary length of 80 mm, and dimensions of 400×400 mm, fixed boundary length of 100 mm under 65 N concentrated force, are compared. The difference method based on the large deflection theory and the difference method based on the small deflection theory are used for theoretical calculation. The difference method is 40×40 units. The simulation process of ABAQUS is calculated by linear analysis and nonlinear analysis with large deformation. As shown in Figures 7 and 8, the curves of the maximum deflection of two kinds of skin

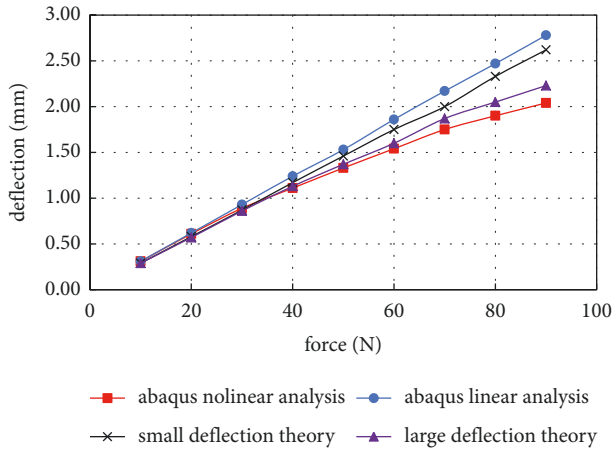


Figure 7: Curve of maximum deflection of 400 × 400 skin changing with concentrated force.

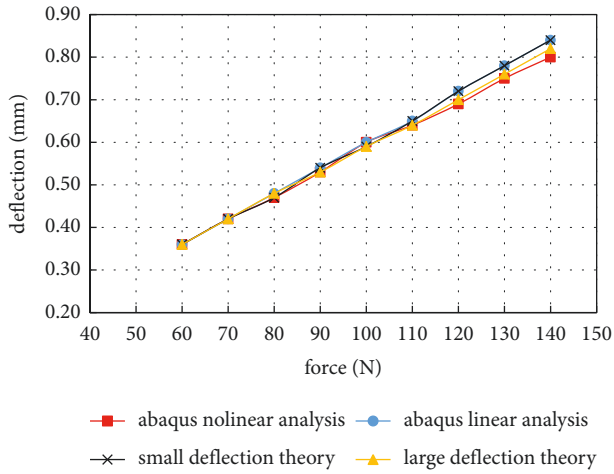


Figure 8: Curve of maximum deflection of 200 × 200 skin changing with concentrated force.

changing with the concentrated force are shown. It can be seen that when the maximum deflection is less than 1 mm (that is, half of the skin thickness), the error of the four calculation results can be controlled within 0.10 mm. With the increase of deflection, the calculation results based on the small deflection theory and ABAQUS linear analysis are approximate, and the maximum error within 2 mm deflection is 5.76%. However, based on the small deflection theory and ABAQUS linear analysis, the maximum error within 2 mm deflection is 28.43%. This is because the strain and internal force caused by the longitudinal displacement of each point in the neutral plane are not taken into account in the calculation based on the small deflection theory. With the increase of deformation, the neutral plane stress and strain caused by longitudinal displacement increase, which makes the calculation based on small deflection theory deviate greatly from the actual situation. The calculated results based on the large deflection theory are similar to the results of ABAQUS nonlinear analysis, and the variation trend is close. The maximum error within 2 mm deflection is 12.70%. If it is considered that the finite element nonlinear analysis results with large deformation are closer to the real situation, the calculation result based on large deflection theory is more accurate in the case of large deformation of the skin.

The deflection of ABAQUS nonlinear analysis results, calculation results based on small deflection theory, and calculation results based on large deflection theory at the red dot position shown in Figures 9–11 are compared. The skin with 400 mm size and 100 mm fixed boundary length is selected to analyze the results under the action of 50 N concentrated force. Figure 9 shows the finite element simulation results, Figure 10 shows the theoretical difference calculation results of large deflection, and Figure 11 shows the theoretical difference calculation results of small deflection. Figure 12 shows the comparison of the three calculation results. It can be seen that the deflection change trend of the three calculation methods is consistent at different node positions, which indicates that the difference method based on thin plate theory not only has more accurate calculation results at the maximum deflection but also can accurately reflect the deflection at different positions of the skin. When the deflection exceeds 1 mm, the calculated results based on the large deflection theory are closer to the results of ABAQUS nonlinear analysis, which is the same as the previous conclusion. By comparison, when the actual deformation is large, the calculation results based on the small deflection theory exaggerate the deformation to a certain extent. At this time, the design and calculation according to the small deflection is safe and wasteful, which is consistent with the conclusion mentioned in the literature [12].

The calculation results based on the large deflection theory are compared under the different grid densities of 20×20, 30×30, 40×40, and 50×50, and the simulation results are shown in Figure 13. In addition to the large deviation of the calculation results when using the difference grid density of 20×20, the deflection of the other three cases keeps a high consistency with the change of load, the maximum deviation is 1.3%, and the deviation is 0.03 mm, which can be ignored. However, with the increase of grid density, the computing time will be doubled. For example, under the same computing conditions, the computing time of 20×20, 30×30, 40×40, and 50×50 differential grid density is 1.5 s, 2.9 s, 8.8 s, and 20.0 s, respectively. This is due to the increase of the unknowns in the difference equations and the capacity of the coefficient matrix with the increase of the grid density. Moreover, under the same conditions, with the increase of deflection, the calculation time will also increase, which is caused by the gradual approximation calculation method of large deflection theory. Under the assumption of the initial value, the initial error of large and small deflection theory is large, so that the number of iterations increases. Therefore, it is necessary to choose the suitable grid density in different cases.

The following conclusions can be obtained by the error evaluation of the difference method above.

(1) The results based on the small deflection theory are similar to the results of the finite element linear

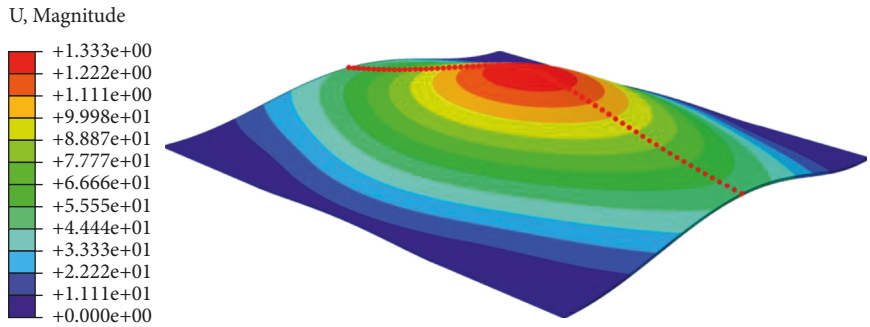


Figure 9: Nephogram of maximum deflection deformation of finite element skin.

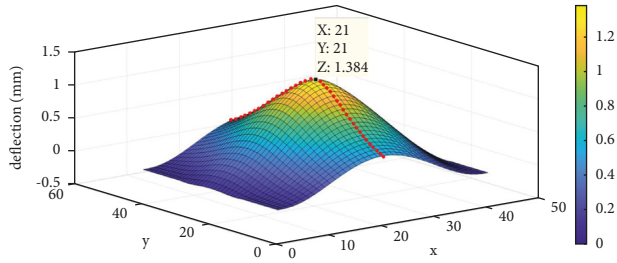


Figure 10: Theoretical calculation results of large deflection.

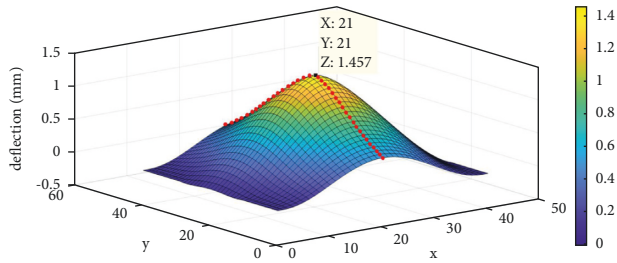


Figure 11: Theoretical calculation results of small deflection.

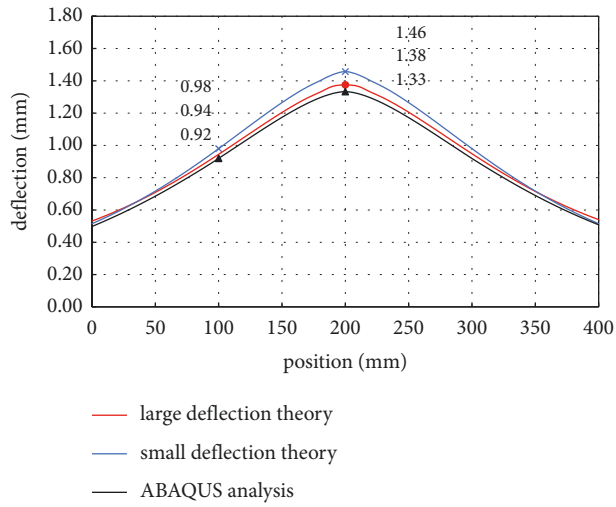


Figure 12: Deflection curve of three calculation methods with node position.

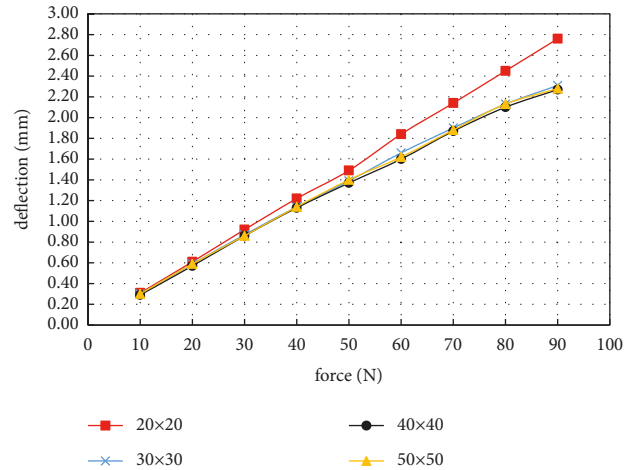


Figure 13: Deflection versus load curves under different grid densities.

analysis, and the results based on the large deflection theory are also similar to the results of the finite element nonlinear analysis. It is proved that the model based on thin plate theory is accurate, and the approximate result of difference calculation is reliable. If it is considered that the results of nonlinear finite element analysis are closer to the real situation, the large deflection theory is more accurate.

(2) When the deflection is less than half of the skin thickness, the calculation results based on the large/small deflection theory are both accurate; When the deflection is more than half of the skin thickness, the calculation results based on the large deflection theory can still approach the accurate value of the simulation, and the calculation results based on the small deflection theory will exaggerate the deformation to a certain extent. At this time, if the design and calculation are carried out according to the small deflection, it is too safe and even wasteful.

(3) When the deflection is near the skin thickness, the calculation results based on the large deflection theory can approach the simulation results, but there is still 12.70% error. The possible error sources include the following points:

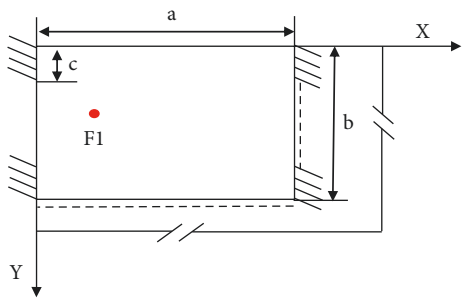


Figure 14: Sketch of skin corner position.

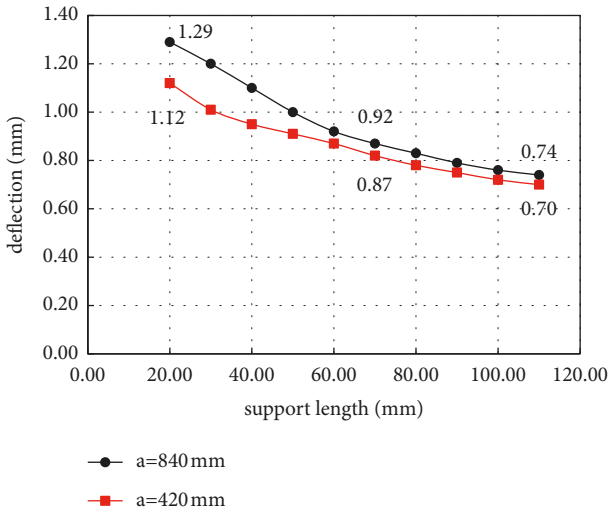


Figure 15: The relationship between the length of the clamping surface at the corner and the maximum deflection of the skin.

(i) The difference method is derived by using the Taylor series expansion of the function and omitted the higher-order terms above the second order, and its error cannot be ignored. Therefore, the more reserved terms and the smaller the mesh size is, the more accurate results can be obtained, but the amount of calculation will increase.

(ii) In the calculation process based on the large deflection theory, the treatment of the boundary conditions of the stress function is relatively rough. In practice, the fixed boundary will be subject to certain longitudinal constraints, and the stress function of the boundary is different from the approximate solution, which leads to errors.

(iii) When using the difference method, the difference of grid density will also affect the calculation results of deflection. Generally speaking, the denser the grid, the higher the calculation accuracy.

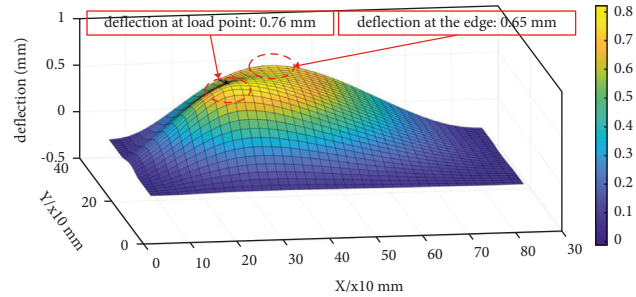


Figure 16: Theoretical difference calculation results of large deflection at corner position.

6. Application Examples

The application verification is carried out by taking the skin corner position of a certain type of aircraft as an example. In Figure 14, a and b represent the edge size of the area and c represents the length of the fixture surface. Study the deflection after applying drilling force at the red point F1, as shown in Figure 14.

Figure 15 shows the relationship between the length of the fixture surface and the maximum deflection of the skin when the two adjacent surfaces of the fixture are effective support segments (b = 280 mm) under the action of 65 N drilling force. The curve of a = 420 mm is added as the control group. It can be seen from Figure 15 that the maximum deflection of the skin decreases with the increase of the length of the board surface. Under the condition of a = 420 mm, the maximum deflection is 1.12 mm and the minimum deflection is 0.70 mm in the range of 20–110 mm. Under the condition of a = 840 mm, the maximum deflection is 1.29 mm and the minimum deflection is 0.74 mm in the range of 20–110 mm. When the length of the surface is more than 60 mm, the deflection difference of the skin is basically maintained within 0.05 mm when the distance between the two plates is different. At this time, the influence of the distance between the two plates on the deflection is very small. This shows that the skin deflection under concentrated load is mainly affected by the boundary condition of the surrounding area. When the area is large enough, the influence of the change of the far boundary on the skin deflection can be almost ignored.

Figure 16 shows the difference calculation results based on the large deflection theory when a = 840 mm and the support length is 100 mm, and Figure 17 shows the finite element analysis results under the same conditions. Comparing the two figures, it can be seen that the deflection error of the theoretical calculation results and the finite element analysis results at the load point is 0.05 mm, and the error of the maximum deflection at the edge is 0.04 mm, and the overall deflection diffusion trend is the same, indicating that the theoretical calculation results are accurate.

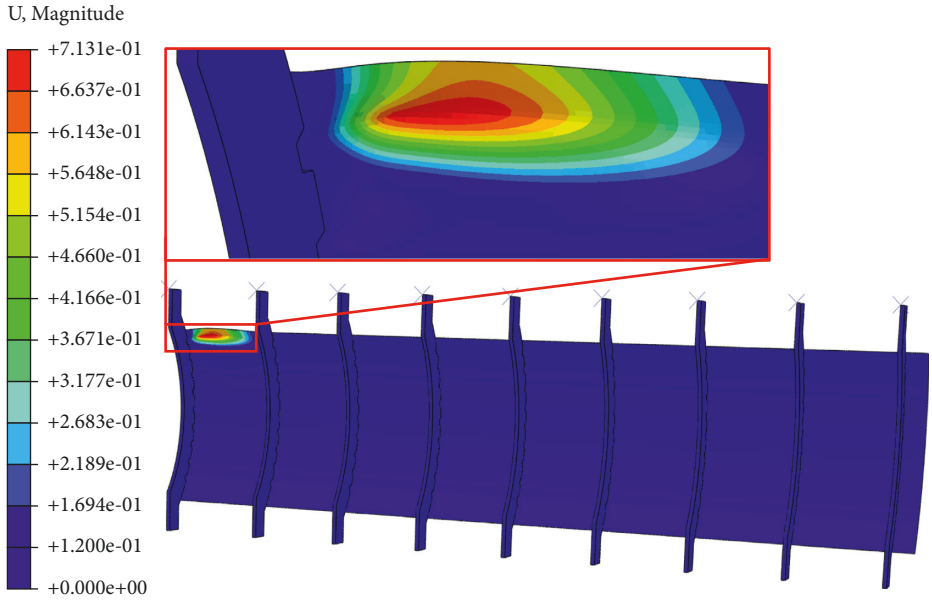


Figure 17: Finite element deformation nephogram of corner position.

7. Conclusions

In this study, aiming at the problem of fixture design in flexible tooling, the load deformation problem of the skin supported by the fixture is transformed into the thin plate bending problem in elastic mechanics, and the mechanical models are established based on the small deflection theory and the large deflection theory, respectively. Considering the complexity of the boundary conditions of the skin under the support of the fixture, the difference method is used to solve the theoretical model. The coefficients of the difference equation are transformed into the form of coefficient matrix, and the approximate values of the deflection of each node of the skin are obtained by matrix operation. The difference results based on small deflection theory and large deflection theory are compared with the finite element simulation results. Finally, the skin corner position of a certain type of aircraft is taken as an example to verify. The support area and support effect of the fixture on the skin can be calculated and optimized by the numerical calculation method proposed in the manuscript, which guarantees the skin deformation within the allowable range when reducing the effective support section of the fixture to the skin. It has important theoretical guiding significance for the flexible design of fixture.

Data Availability

The data, models, and code generated or used to support this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to thank the support of the National Natural Science Foundation of China (52005436) and Science and Technology Innovation 2025 Major Project of Ningbo (2019B10080 and 2020Z068).

References

[1] H. Cai, J. Zhu, and W. Zhang, "Quality deviation control for aircraft using digital twin," *Journal of Computing and Information Science in Engineering*, vol. 21, no. 3, Article ID 031008, 2021.

[2] A. Rezaei Aderiani, K. Wärmefjord, R. Söderberg, and L. Lindkvist, "Individualizing locator adjustments of assembly fixtures using a digital twin," *Journal of Computing and Information Science in Engineering*, vol. 19, no. 4, Article ID 041019, 2019.

[3] W. Cai, S. J. Hu, and J. X. Yuan, "Deformable sheet metal fixturing: principles, algorithms, and simulations," *ASME. J. Manuf. Sci. En*, vol. 118, no. 3, pp. 318–324, 1996.

[4] M. D. Do, Y. Son, and H. J. Choi, "Optimal workpiece positioning in flexible fixtures for thin-walled components," *Computer-Aided Design*, vol. 95, pp. 14–23, 2018.

[5] C. Chen, Y. Sun, and J. Ni, "Optimization of flexible fixture layout using N-M principle," *International Journal of Advanced Manufacturing Technology*, vol. 96, pp. 4303–4311, 2018.

[6] A. Masoumi and J. S. Vahid, "Fixture layout optimization in multi-station sheet metal assembly considering assembly sequence and datum scheme," *International Journal of Advanced Manufacturing Technology*, vol. 95, no. 9-12, pp. 4629–4643, 2018.

[7] Z. H. Ma, Y. F. Xing, and M. Hu, "Fixture layout optimization based on hybrid algorithm of gaot and rbf-nn for sheet metal parts," in *Proceedings of the 2019 International Conference on Artificial Intelligence and Advanced Manufacturing*, pp. 1–6, Dublin, Ireland, October 2019.

[8] S. E. Zhou, Q. Chan, Z. Y. Liu, and J. R. Tan, "A rapid design method of anti-deformation fixture layout for thin-walled structures," in *Proceedings of the International Conference on Mechanical Design*, pp. 721–733, Vancouver, British Columbia, Canada, August 2017.

[9] X. J. Wan, J. Q. Yang, H. J. Zhang, Z. Y. Feng, and Z. Xu, "Optimization of fixture layout based on error amplification factors," *Journal of Computing and Information Science in Engineering*, vol. 18, no. 4, pp. 41001–41007, 2018.

[10] Y. Yang, Z. Q. Wang, B. Yang, Z. Jing, and Y. Kang, "Multi-objective optimization for fixture locating layout of sheet metal Part Using SVR and NSGA-II," *Mathematical Problems in Engineering*, vol. 2017, Article ID 7076143, 10 pages, 2017.

[11] A. Zeshan, T. Sultan, M. Asad, M. Zoppi, and R. Molfino, "Fixture layout optimization for multi point respot welding of sheet metals," *Journal of Mechanical Science and Technology*, vol. 32, no. 4, pp. 1749–1760, 2018.

[12] C. Li, Z. Q. Wang, H. Tong, and S. Tong, "Design of fixture locating layout for thin-walled part based on Kriging and FPA," *Aeronautical Manufacturing Technology*, vol. 63, no. 18, pp. 95–101, 2020.

[13] Z. Q. Wang, J. Huang, Y. G. Kang, B. Yang, and Y. Yang, "Locating strategy optimization of aircraft weakly rigid parts assembly based on firefly algorithm," *Mechanical Science and Technology for Aerospace Engineering*, vol. 035, no. 4, pp. 626–629, 2016.

[14] X. N. Li, Y. S. Wang, and Y. H. Li, "Optimization design of aircraft weak rigid parts clamping scheme based on genetic algorithm," *Aeronautical Manufacturing Technology*, vol. 62, no. 1, pp. 82–86+94, 2019.

[15] A. Alibeigloo, "Static analysis of functionally graded carbon nanotube- reinforced composite plate embedded in piezo-electric layers by using theory of elasticity," *Composite Structures*, vol. 95, pp. 612–622, 2013.

[16] Z. B. Kuang, "An applied electro-magneto-elastic thin plate theory," *Acta Mechanica*, vol. 225, no. 4, pp. 1153–1166, 2014.

[17] D. J. Steigmann, "Thin-plate theory for large elastic deformations," *International Journal of Non-linear Mechanics*, vol. 42, no. 2, pp. 233–240, 2007.

[18] Z. L. Xu, *Elasticity*, Higher Education Press, Beijing, China, 2016.

[19] J. Sladek and V. Sladek, "A meshless method for large deflection of plates," *Computational Mechanics*, vol. 30, no. 2, pp. 155–163, 2003.

[20] A. A. Shabana, *Dynamics of Multibody Systems*, Cambridge University Press, Cambridge, UK, 2003.

[21] F. Chen, B. Chen, S. Chen, and Q. Yang, "Machining Error Compensation Method for Thin-Walled Parts Machining Process Based on Bicubic B-Spline Interpolation," *Aeronautical Manufacturing Technology*, vol. 59, no. 4, pp. 63–67, 2016.

[22] X. Li, J. T. Yuan, Z. H. Wang, and B. Zhang, "Study on deformation control of Thin-walled titanium alloy parts in non-uniform allowance machining based on Rayleigh-Ritz method," *China Mechanical Engineering*, vol. 31, no. 11, pp. 1378–1385, 2020.

[23] Z. C. Tang, Z. J. Fu, and J. M. Fan, "Generalized finite difference method for solving Kirchhoff plate and Winkler plate bending problems," *Chinese Journal of Solid Mechanics*, vol. 39, no. 4, pp. 419–428, 2018.

[24] M. R. Bahubalendruni and B. B. Biswal, "A review on assembly sequence generation and its automation," *Proceedings of the Institution of Mechanical Engineers - Part C: Journal of Mechanical Engineering Science*, vol. 230, no. 5, pp. 824–838, 2016.

[25] Y. F. Yang, M. Yang, L. Shu, S. Li, and Z. Liu, "A novel parallel assembly sequence planning method for complex products based on psobc," *Mathematical Problems in Engineering*, vol. 2020, Article ID 7848329, 11 pages, 2020.

[26] M. R. Bahubalendruni and B. B. Biswal, "An intelligent approach towards optimal assembly sequence generation," *Proceedings of the Institution of Mechanical Engineers - Part C: Journal of Mechanical Engineering Science*, vol. 232, no. 4, pp. 531–541, 2016.

[27] M. V. A. Bahubalendruni, A. K. Gulivindala, S. S. V. Varupala, and D. K. Palavalasa, "Optimal assembly sequence generation through computational approach," *Sādhanā*, vol. 44, no. 8, pp. 1–9, 2019.

[28] Z. L. Xu, *Elasticity*, Higher Education Press, Beijing, China, 2016.

[29] S. G. Liu and T. Zhang, *Basic Theory of Elasticity and Plasticity*, Huazhong University of Science and Technology Press, Wuhan, China, 2008.

[30] B. M. Edward, *An Engineer's Guide to MATLAB*, Publishing House of Electronics Industry, Beijing, China, 2006.

Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 6872162, 13 pages
https://doi.org/10.1155/2022/6872162

Research Article

Improved Hypercube Optimisation Search Algorithm for Optimisation of High Dimensional Functions

Mustafa Tunay[1] **and Rahib Abiyev**[2]

[1]*Department of Computer Engineering, Istanbul Gelisim University, Istanbul, Turkey*
[2]*Department of Computer Engineering, Near East University, Lefkosa, Northern Cyprus, Mersin 10, Turkey*

Correspondence should be addressed to Rahib Abiyev; rahib.abiyev@neu.edu.tr

Received 10 September 2021; Revised 28 November 2021; Accepted 10 March 2022; Published 22 April 2022

Academic Editor: Guoqiang Wang

This paper proposes a stochastic search algorithm called improved hypercube optimisation search (HOS+) to find a better solution for optimisation problems. This algorithm is an improvement of the hypercube optimisation algorithm that includes initialization, displacement-shrink and searching area modules. The proposed algorithm has a new random parameters (RP) module that uses two control parameters in order to prevent premature convergence and slow finishing and improve the search accuracy considerable. Many optimisation problems can sometimes cause getting stuck into an interior local optimal solution. HOS+ algorithm that uses a random module can solve this problem and find the global optimal solution. A set of experiments were done in order to test the performance of the algorithm. At first, the performance of the proposed algorithm is tested using low and high dimensional benchmark functions. The simulation results indicated good convergence and much better performance at the lowest of iterations. The HOS+ algorithm is compared with other meta heuristic algorithms using the same benchmark functions on different dimensions. The comparative results indicated the superiority of the HOS+ algorithm in terms of obtaining the best optimal value and accelerating convergence solutions.

1. Introduction

The optimisation includes finding the best solutions in a solution space for which the objective function obtains its smallest (or largest) value. Real-world optimisation problems are often nonlinear and can have multiple local optimal (minimum and maximum) solutions. The basic aim is to find the best of these local optimums. Generally, global optimisation includes finding the best available solution from all feasible solutions given in a defined domain for which the objective function will obtain its smallest (or largest) value.

Traditional numerical optimisation algorithms that are based on finding the derivative of the objective function cannot find global optimal points for the function having multiple local optimums. In such cases, one efficient approach is based on the use of heuristic search algorithms. Metaheuristic search algorithms that are based on directed random search methods can provide sufficiently good solutions and solve the local-optimum problem and find global solutions to the optimisation problems [1, 2]. A set of meta-heuristic optimisation algorithms is developed to find the best solutions. These algorithms are: bat algorithm (BAT) [3], cuckoo search (CS) [4], ant lion optimizer (ALO) [5, 6], elephant herding optimisation (EHO) [7, 8], moth-flame optimisation (MFO) [9], krill herd (KH) [10], moth search algorithm (MSA) [11], monarch butterfly optimisation (MBO) [12, 13], mussels wandering optimisation (MWO) [14], and whale optimisation algorithm (WOA) [15]. Other meta-heuristic optimisation algorithms such as differential evolution (DE) [16], biogeography-based optimisation (BBO) [17, 18], harmony search (HS) [19], evolution strategies (ES) [20], sine cosine algorithm (SCA) [21], gravitational search algorithm (GSA) [22], monkey algorithm [23, 24], dragonfly algorithm (DA), and hybrid ABC/DA (HAD) [25, 26] are also efficiently used for solving many optimisation problems.

Paper [27] proposed an intelligent swarm-based MBO algorithm inspired by the migration behavior of monarch butterflies in nature. In this algorithm, the whole population is partitioned into two subpopulations of equal size. Each individual in population 1 changes its position based on the migration operator, each individual in population 2 changes its position according to the butterfly adjusting operator. The algorithm contains exploration and exploitation properties, easy structure, and strong robustness and is designated for global optimisation. The paper [28] proposed a Slime mould algorithm that is based on the oscillation mode of slime mould in nature. The algorithm uses adaptive weights to simulate the process of producing positive and negative feedback of the propagation wave of slime mould based on bio-oscillator to form the optimal path for connecting food with the excellent exploratory ability and exploitation property. In the moth search algorithm [29], the best moth is viewed as a light source. Some neighbour moths that are close to fittest always display an inclination to fly around their own positions in the form of Levy flights. In contrast, the moths that are far from the fittest one will fly towards the best one in a big step. These two operations corresponding to exploration and exploitation are the basis of the MSO algorithm. The paper [30] proposed a population based hunger game search. The algorithm is based on hunger-driven activities and behavioural choice animals. The authors used the algorithm for different areas such as artificial intelligence and machine learning with high optimisation capacity. The paper [31] based on the predation of animals proposed a colony predation search algorithm. The algorithm utilizes mathematical modelling of animal hunting. The algorithm was used for solving engineering problems. The paper [32] proposed a mathematical optimisation model based on simulation of the hunting behavior of Harris hawks. Inspired by the cooperative behavior and chasing style of Harris hawks, the authors designed the algorithm. A number of benchmark examples were used to evaluate the performance of the algorithm.

As mentioned, many metaheuristic optimisation algorithms have been designed to find the best solution to global problems and increase the accuracy of the optimisation. However, the optimisation algorithms can sometimes get stuck into an interior local optimal solution and cannot escape from that state. These search algorithms have premature convergence problems and low search accuracy in solving optimisation problems. This happens due to the loss of diversity among individuals. The original HOS algorithm may also have the same problems and this can lead to a lack of finding a near-optimal solution in the search area. In this paper, we proposed a new version of the HOS algorithm. The novelties of this paper are: The novel structure of the HOS+ algorithm is proposed; The new random perturbation module of the HOS+ algorithm is introduced; The proposed algorithm has been tested on benchmark problems; The proposed HOS+ algorithm help to prevent premature convergence problems and find the best solutions and also improve search accuracy in a small number of iterations. The designed HOS+ algorithm provided passing over possible local optima and has proven to be a successful convergence optima solution for the lowest iterations.

The remainder of the paper is organised as follows: Sec.2 presents the improved HOS+ algorithm. The design stages and operation modules of HOS+ are explained. Sec.3 presents the experimental results and discussion. A set of benchmark functions of different dimensions was used for testing the proposed HOS+ algorithm. In Sec.4 the performance of the HOS+ algorithm is evaluated and compared with the performances of other meta-heuristic algorithms using the same test functions of different dimensions. Finally, the conclusion is presented in Sec 5.

2. HOS+ Algorithm

The improved HOS+ algorithm is a new stochastic search method inspired by a hypercube evolution. The algorithm is a derivative-free unconstrained optimisation method and is based on a set of points randomly distributed inside an m-dimensional hypercube. The proposed algorithm provides the movement of population (number of points inside of the hypercube) that reaches the minimum (or maximum) of objective function rapidly by reducing the area of the hypercube and updating and searching solutions at each iteration. The original HOS algorithm consists of three-initialization process, displacement- shrink process and searching areas process. The proposed algorithm is renewed by adding a random module in the original search processes.

Stochastic processes are mathematical models of systems that are changing randomly. They are characterised by random variables described by a random probability distribution. They have applications in different fields such as physics, industry, economy, information technology, computer science and many other fields. There are two ways to use a random process in an optimisation problem: through a cost function or a set of constraints. At the same time, stochastic optimisation also refers to any optimisation technique that uses randomness in some ensembles. We consider the case where the parameters of objective function or constraints are random. The improved HOS+ algorithm is inspired by a random process and uses random parameter $p1$ and parameter $p2$ in order to improve the problems of premature convergence and slow finishing and search accuracy considerable. The proposed algorithm is the improvement of the stochastic hypercube optimisation algorithm (HOS) algorithm presented in [1, 2]. The algorithm is presented in Figure 1 and explained by the following steps in detail.

2.1. Step A: Initialization Process. The initialization process starts by generating initial points and forms the initial matrices for evaluating solutions in a given hypercube. The initial points are generated using the following operations.

(1) Initialize the solution using the dimension of hypercube.

$$\mathbf{X} = \text{random}\,(n, m), \tag{1}$$

where m is the dimension of the hypercube, n is population size.

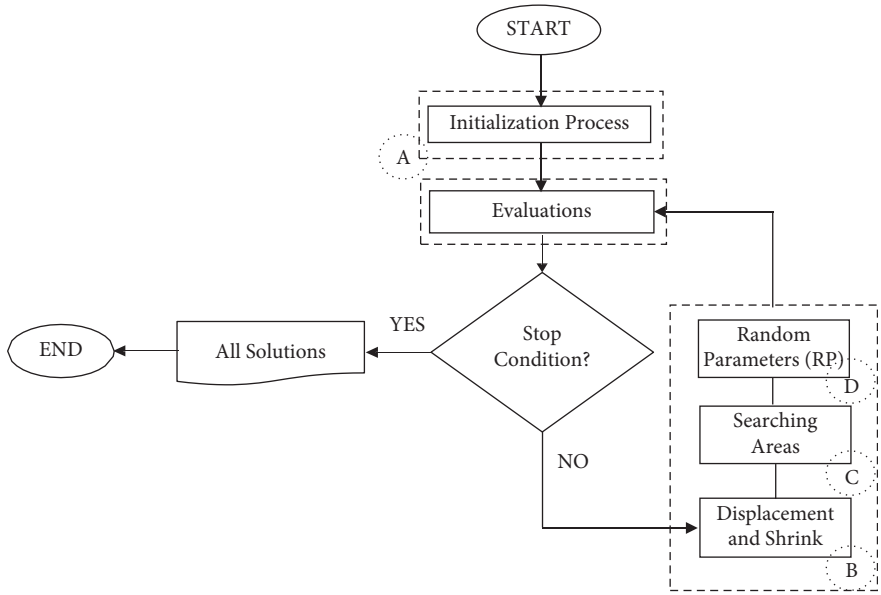


Figure 1: Flowchart of the improved HOS+ algorithm.

(2) Use lower bound (LB) and upper bound (UB) to scale the solutions x_{ij}.

$$x_{ij} = \text{LB} + x_{ij}\,(\text{UB} - \text{LB}). \tag{2}$$

(3) Find the radii (R) of hypercube.

$$R = \text{UB} - \text{LB}. \tag{3}$$

(4) Find the center of hypercube X_c.

$$X_c = R/2, \tag{4}$$

Formula (1) initialize the solutions X inside hypercube which is search area.

$\mathbf{X} = \begin{bmatrix} \mathbf{x}_{1,1} & \cdots & \mathbf{x}_{1,\mathbf{m}} \\ \vdots & \ddots & \vdots \\ \mathbf{x}_{\mathbf{n},1} & \cdots & \mathbf{x}_{\mathbf{n},\mathbf{m}} \end{bmatrix}$, where m is the dimension of the hypercube, n is population size. Each position is evaluated using an objective function. The best point X_{best} is determined according to the values of test (or objective) function **F**.

In the initialization stage, the initial solutions are generated using initial conditions such as the dimension of the hypercube (m), radii of the hypercube (**R**), lower-upper boundaries (LB, UB) and a number of points (population, m) inside the hypercube (Figure 2). The lower and upper boundaries are used to generate the hypercube. The basic parameters of the hypercube are the center X_c and radii R, which are formulated by formulas (3) and (4). In the given search interval, using generated $x_{ij}(i = 1,\ldots,n;\ j = 1,\ldots,m)$ data points inside the hypercube, the values of the objective functions f_{ij} are calculated (here f_{ij} are elements of **F**). After each iteration, the points change their positions (movement). These initial points are evaluated according to the objective function. So initialization process creates matrices as X_{best}, F_{best} ($nx1$) after evaluating initial points. The

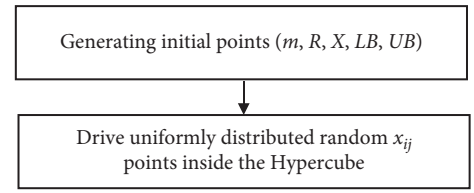


Figure 2: The steps of the initialization process.

determined X_{best} point is improved (updated) using local searches, such as hill climbing or derivative-based local search. If we use a derivative-based local search then $X_{\text{best}}^{\text{new}} = X_{\text{best}} + \rho\nabla F$, where $0 \le p \le 1$, F is the objective function. The details of the initialization process in the HOS+ algorithm are shown in Figure 2. In the next iteration, the X_{best} is utilised to determine the Hypercube center. This operation is realised by computing the center and mean of the last position point (X_c) and the last best X_{best} point. The given process is called the "displacement" process.

2.2. Step B: Displacement–Shrink Process. The displacement–shrink process determines the hypercube's centre and evaluates the test (or objective) function. The centre of the next hypercube is evaluated using the average of the sum of the previous hypercube's centre and the present best point (X_{best}). Thus the centre of the next hypercube (new hypercube) is determined as

$$X_{\text{new\_center}} = \frac{X_c + X_{\text{best}}}{2}, \tag{5}$$

$$R_{\text{new}} = R * S. \tag{6}$$

Here R and R_{new} are old and new radii, S is the convergence factor calculated in Section 3. The updates of hypercube parameters are performed using (5) and (6). As a

result of this process, the hypercube size and correspondingly the search space are reduced. The process is called "shrink." The decrease in hypercube size leads to an increase in the density of the search points (population). The movement of the best value is governed by contraction. The contraction is greater for smaller movements. This guarantees fast convergence, while it protects against getting stuck at an undesired (local) minimum.

As shown, new data points are generated at each iteration and the objective function is evaluated. According to the evaluation results, the hypercube size is changed. As a result, the hypercube size is decreased and the search space is shrunk correspondingly. The decrease of hypercube size causes an increase in the density of test points. This process causes a rapid finding of the optimum value of the objective function.

The algorithm will pass through a series of points from the current position which determines the maximal distance. The displacement ranges are presented below.

(1) Normalized x_{ij}:

$$x_{ij}^{n} = \frac{\left(x_{ij} - X_{c}\right)}{R}. \tag{7}$$

x_{ij}^{n} is a normalized value of $^{ij}_{x}$.

(2) Normalized X_{best}:

$$X_{\text{best}}^{n} = \frac{\left(X_{\text{best}} - X_{c}\right)}{R}. \tag{8}$$

X_{best}^{n} is the normalized value of X_{best}.

(3) Normalize distance d_n:

$$d_{n} = \frac{\left(\text{sum}\left(x_{ij}^{n} - X_{\text{best}}^{n}\right)^{2}\right)^{1/2}}{R}. \tag{9}$$

(4) Re-normalize distance:

$$d_{nn} = \frac{d_{n}}{\sqrt{m}}. \tag{10}$$

The x displacement is calculated and normalized twice for each iteration: at first, each element of x is divided by its corresponding initial interval so that the displacement is converted into unity-sided points (equations (7) and (8)), and then this number is again normalized by dividing it to the diagonal of the points such as $\sqrt{m}$ (equations (9) and (10)). Thus, the contraction of the hypercube is becoming higher, when the movement of the number of points shrink which accelerates the convergence.

2.3. Step C: Searching Areas Process. Using equations (7)–(10) the distances between new and old optimum values are calculated in this process. In addition, the "Searching areas" process uses the interval defined for re-normalized distance, given in (11), to control the movements of x.

$$0 \le dm \le 1. \tag{11}$$

In case of satisfaction the condition by the movement x, the convergence factor S is computed and updated at each iteration as

$$S = 1 - 0.2e^{-3d_{nn}}. \tag{12}$$

In the above equation, d_{nn} is the normalized distance calculated by (10) and based on the average of the last two best values of x. Thus, the purpose of the proposed algorithm ensures the movement of the population that reaches the minimum point rapidly by reducing the area of the hypercube after each iteration. A flowchart of the searching areas process in the proposed algorithm is shown in Figure 3.

2.4. Step D: Random Parameter (RP) Module. HOS+ algorithm includes a new RP module characterised by two control parameters *p1* and *p2*. This module improves the points (current positions) inside the hypercube that might get stuck at some local solutions. At first, the *p1* improves the points having local optima problem. The process is continued according to some tolerance and fixed by *tolX*. In addition, the upper bound of dimension d can be determined according to the value of *tolX*. The value of the solution at the local point is updated by multiplying the parameter p1 by a random scalar drawn from the standard normal distribution, that is $X^{\text{new}} \leftarrow X*(1 + p1*\text{rand}n)$. Here rand$n$ generates random numbers in the interval of 0 and 1. The new point will be accepted according to the values of test functions. If the value of the optimisation function in the new point will be minimum (or maximum) than the previous one then the new point will be included in the solution. Thus, by using this operation the presented random module prevents the point from getting stuck in some local optimal solutions while controlling the points' positions inside the hypercube. After these operations, the second new random parameter is introduced in order to control directed movements of all points inside the hypercube. The introduced second parameter ($p2$) improves the solutions along the direction pointing to their current position with different perturbations originating from some possible local minimum and searching for another minimum point. The points are updated by multiplying the parameter p2 to uniformly distributed random numbers (rand [1 x D]). That is

$$X^{\text{new}}(\cdot) \leftarrow X(\cdot) * (1 + p2 * \text{rand}[1, D]). \tag{13}$$

where D is searching dimension. Thus, the improvements of positions are performed along the direction pointing with different perturbations in order to exit from some possible local minimum or to search for another minimum point. The pseudocode of the random permutation module is given in Figure 4.

The computational complexity of the HOS+ algorithm was analyzed. HOS+ includes initialization, fitness evaluation, displacement-shrink and normalization, searching areas and random parameter modules. Hypercube dimensions n, population size m and a maximum number of iterations T are the main parameters affecting the running time of the HOS+ algorithm in these modules.

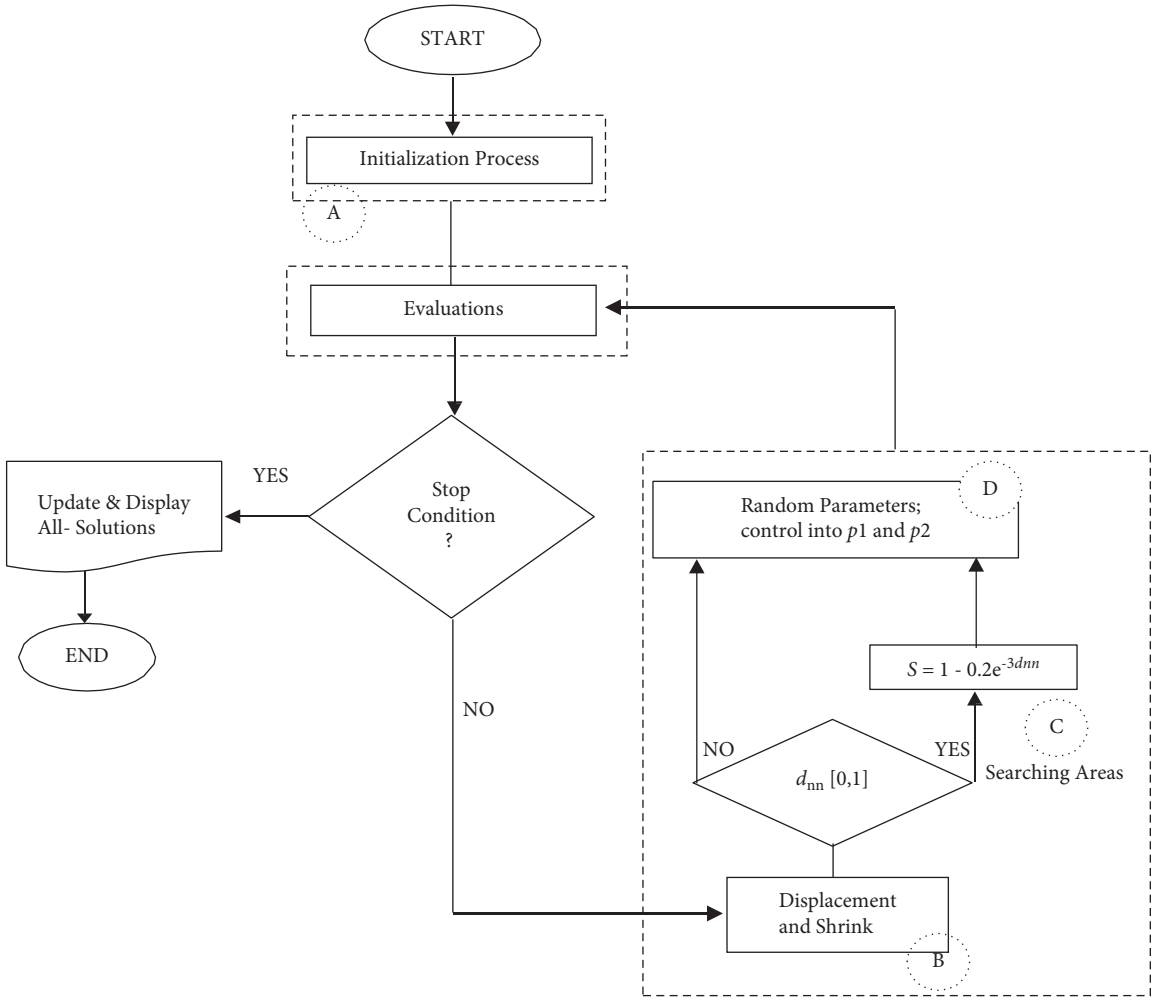


Figure 3: Flow chart of the searching areas process in the proposed algorithm.

Computational complexity of initialization is $O(n{\cdot}m)$, displacement- shrink and normalization module is $O(n{\cdot}m)$, searching areas $O(n)$, random permutation module is $O(n{\cdot}m)$. The displacement-shrink and normalization module, searching areas, random permutation module running in each iteration t; if we take into account the maximum number of iterations T then the computational complexity of HOS+ will be presented by $O(n{\cdot}m + n{\cdot}T + n{\cdot}m{\cdot}T)$.

3. Benchmark Functions

The benchmark functions used are Sphere function (F1), Schwefel 2.22 function (F2), Rotated Hyper-Ellipsoid function (F3), Ackley function (F4), Griewank function (F5), and Hyperellipsoid function (F6). The details of information for these test functions are given below. The performance of the HOS+ was evaluated using low, medium and high dimensional optimisation functions. In the paper, low dimension is taken equal to 30D, medium dimension equal to 60D and high dimension-90D. For more information about these benchmark functions, we refer the reader to the link: https://www.sfu.ca/~ssurjano/optimisation.html.

3.1. Sphere Function (F1). The function is convex, continuous, differentiable, separable and uni-modal. It is used $x_i \in [-5.12, 5.12]$ for all $i = 1,\ldots,n$ and the global minimum is at $f(x) = 0$.

$$f(x) = f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} x_i^2. \tag{14}$$

3.2. Schwefel 2.22 Function (F2). The function is convex, continuous, non-differentiable, separable and uni-modal. It is used $x_i \in [-10,10]$ for all $i = 1,\ldots,n$ and the global minimum is at $f(x) = 0$.

$$f(x) = f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i|. \tag{15}$$

3.3. Rotated Hyper Ellipsoid Function (F3). This function is convex, continuous and uni-modal. It is used $x_i \in [-65, 65]$ for all $i = 1,\ldots,n$ and the global minimum is at $f(x) = 0$.

```
An RP module (→ Step D) of the HOS+ algorithm
Input (Step D): Number of Population (n), Number of Tolerance (tolX), Lower
Bound (LB) and Upper Bound (UB), Random Parameters (p1 and p2), randn [0,1],
Global maximum is searched (min)
Output: Best Solution (Best (X))
for i = 1 to n do
    for j = 1 to tolX do
        d ← ceil (rand x D)
        X_i^new ← X_id * (1 + p1 * randn)
        if LB < X_i^new < UB then
            Continue
        end if
        if min > 0 then
        | if f(X_i^new) > f(X_ij) then
            | X_ij ← X_i^new
        | end if
        Else
        | if f(X_i^new) < f(X_ij) then
        |   | X_ij ← X_i^new
        | end if
        end if
    end for
end for
for i = 1 to n do
    for j = 1 to tolX do
        X_i^new ← X_ij (·) (1 + p2 * rand [1, D])
        if LB < X_i^new < UB then
            Continue
        end if
        if min > 0 then
        | if f(X_i^new) > f(X_ij) then
        |   | X_ij ← X_i^new
        | end if
        Else
        | if f(X_i^new) < f(X_ij) then
        |   | X_ij ← X_i^new
        | end if
        end if
    end for
end for
Output
```

Figure 4: Pseudocode of random perturbation module.

$$f(x) = f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} \sum_{j=1}^{i} x_j^2. \tag{16}$$

3.4. Ackley Function (F4). This function is continuous and multi-modal. It is used $x_i \in [-32, 32]$ for all $i = 1,\ldots,n$ and the global minimum is at $f(x) = 0$.

$$f(x) = f(x_1, x_2, \ldots, x_n) = -20 \exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos(2\pi x_i)\right) + 20 + e. \tag{17}$$

3.5. Griewank Function (F5). This function is continuous and uni-modal. It is used $x_i \in [-600, 600]$ for all $i = 1,\ldots,n$ and the global minimum is at $f(x) = 0$.

$$f(x) = f(x_1, x_2, \ldots, x_n) = 1 + \sum_{i=1}^{n} \frac{x_i^2}{4000} - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right). \tag{18}$$

3.6. Hyper Ellipsoid Function (F6). This function is convex, continuous, differentiable, separable and uni-modal. It is used $x_i \in [-5.12, 5.12]$ for all $i = 1,\ldots,n$ and the global minimum is at $f(x) = 0$.

$$f(x) = f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} i^2 x_i^2. \tag{19}$$

Table 1: The performances of the HOS and HOS+ algorithms using different dimensions.

Benchmark functions	D	HOS Best	HOS Mean	HOS Std. Dev.	HOS+ Best	HOS+ Mean	HOS+ Std. dev.
Sphere (F1)	30	$1.19E-06$	$8.88E+00$	$4.86E+00$	$9.95E-137$	$2.63E-51$	$4.29E-51$
	60	$1.30E-02$	$5.92E+01$	$4.00E+01$	$7.79E-96$	$6.39E-39$	$6.19E-39$
	90	$1.20E-01$	$1.49E+02$	$7.49E+01$	$1.70E-81$	$2.19E-32$	$3.59E-32$
Schwefel 2.22 (F2)	30	$3.00E-01$	$2.67E+01$	$1.41E+01$	$1.19E-95$	$1.98E-27$	$3.19E-27$
	60	$6.50E-01$	$1.01E+02$	$4.40E+01$	$3.21E-72$	$3.88E-22$	$6.39E-22$
	90	$4.75E+00$	$1.67E+02$	$7.66E+01$	$2.49E-59$	$5.72E-18$	$6.91E-18$
Rotated hyper ellipsoid (F3)	30	$2.59E+02$	$2.10E+04$	$1.11E+04$	$3.39E-134$	$2.69E-47$	$2.03E-47$
	60	$4.08E+02$	$2.77E+05$	$1.59E+05$	$5.25E-109$	$1.75E-34$	$2.09E-34$
	90	$5.85E+02$	$9.81E+05$	$5.09E+05$	$1.05E-75$	$1.35E-28$	$1.79E-28$
Ackley (F4)	30	$5.00E-03$	$1.06E+01$	$3.49E+00$	$8.88E-16$	$3.57E-15$	$4.69E-15$
	60	$5.40E-01$	$1.51E+01$	$3.86E+00$	$8.88E-16$	$2.53E-14$	$2.79E-14$
	90	$5.70E-01$	$1.69E+01$	$4.40E+00$	$4.44E-15$	$4.29E-13$	$6.69E-13$
Griewank (F5)	30	$5.00E-02$	$3.10E+01$	$1.85E+01$	$0.00E+00$	$0.00E+00$	$0.00E+00$
	60	$6.90E-01$	$1.50E+02$	$1.19E+02$	$0.00E+00$	$0.00E+00$	$0.00E+00$
	90	$1.20E+01$	$4.19E+02$	$2.39E+02$	$0.00E+00$	$0.00E+00$	$0.00E+00$
Hyper ellipsoid (F6) sphere (F1)	30	$5.10E+01$	$2.61E+02$	$4.59E+01$	$1.55E-139$	$2.19E-48$	$1.77E-48$
	60	$6.30E+01$	$2.75E+03$	$6.49E+02$	$4.35E-113$	$3.39E-36$	$3.59E-36$
	90	$6.90E+01$	$1.04E+04$	$2.45E+03$	$4.89E-79$	$2.99E-29$	$3.96E-29$

4. The Performance of HOS+ Algorithm on Benchmark Functions

The HOS+ algorithm is simulated in Matlab R2017a for finding optimal solutions for a set of benchmark functions. The computer used for simulations has the following characteristics;

(i) CPU: i5-8250U

(ii) CPU Speed:1.60 GHz–1.80 GHz

(iii) RAM: 8.00 GB

(iv) OS: Windows 10

The HOS+ algorithm has been tested using the above-mentioned benchmark functions on 30D, 60D, and 90D dimensions. Evaluations are carried out using the same population size of 50, the same number of iterations of 50 and the maximum function evaluation. For all cases, the results are averaged using 100 independent runs of the algorithm. For measuring the performances of the algorithm the best, mean and standard deviation are taken.

At first, the performance of the HOS+ algorithm is compared with the original HOS algorithm given in [1, 2]. Using both algorithms the experiments were conducted for all benchmark functions on 30, 60 and 90 dimensions. Table 1 depicts the results of experiments obtained for six optimisation functions of F1, F2, F3, F4, F5, and F6 for the HOS and HOS+ algorithms. The simulations have been done at the same initial conditions. The best, averaged values of mean and standard deviation are illustrated in the table. The convergence plots and time-spent of HOS+ algorithm obtained from the simulations on 90-dimensional optimisation functions F1, F2, F3, F4, F5, and F6 were depicted in Figures 5 to 10, correspondingly. The results of the experiments were presented using the convergence plots and global search ability of the proposed algorithm. For comparative purpose, the convergence plots of original HOS algorithm are presented in Figure 11. The comparative results of performances given in Table 1 and the convergence plots given in Figures 5 to 10 and Figure 11 demonstrate the superiority of the HOS+ algorithm over the original HOS algorithm.

5. Comparison of HOS+ with Other Metaheuristic Algorithms

The HOS+ algorithm performance was compared with the performances of other meta-heuristic optimisation algorithms using 6 test functions on different dimensions, particularly 30D, 60D, and 90D. The comparisons of the algorithms were done using the same initial conditions. All algorithms are simulated using the same iterations' number, the same dimensions, and the same maximum function evaluation [26, 33].

The comparative results of each function are presented in Tables 2–5. The best results are marked in bold. Tables 2–4 depict comparative results of experiments obtained for optimisation functions (F1–F6) on dimensions 30, 60 and 90.100 independent runs have been done for each optimisation function using HOS+ algorithm. Table 5 demonstrates the experimental comparative results of the F1, F2, and F4 functions on dimensions 20, 50, and 100. The results are averaged values of 30 independent runs of each algorithm.

The initial values of the parameters for the HOS+ algorithm were set as follows: population size is set equal to 50 and a number of iterations is set equal to 50.

First, the proposed algorithm was compared with a selected collection of other meta-heuristic algorithms. DA, ABC, and HAD algorithms were taken for comparison. Table 2 illustrates the best, mean and best standard deviation obtained from the experiments.

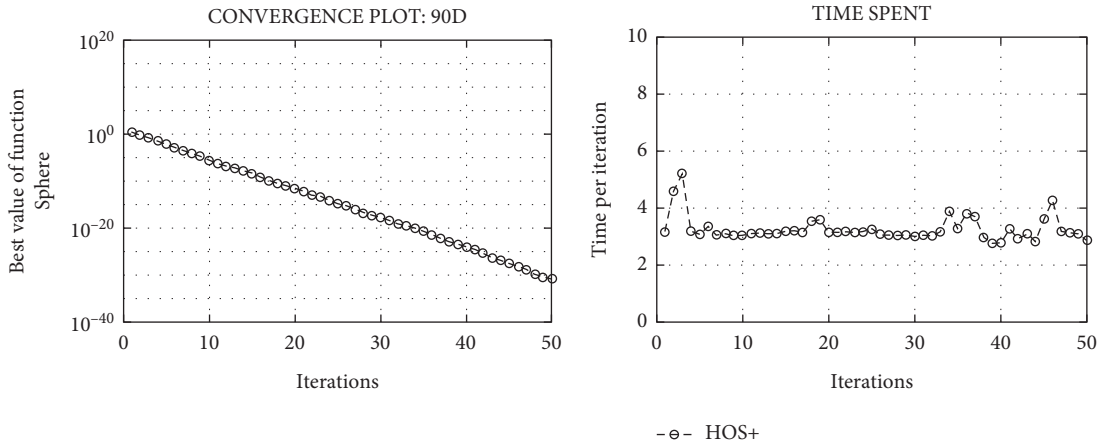


Figure 5: Convergence plot with time spent against 90-dimensional Sphere function (F1).

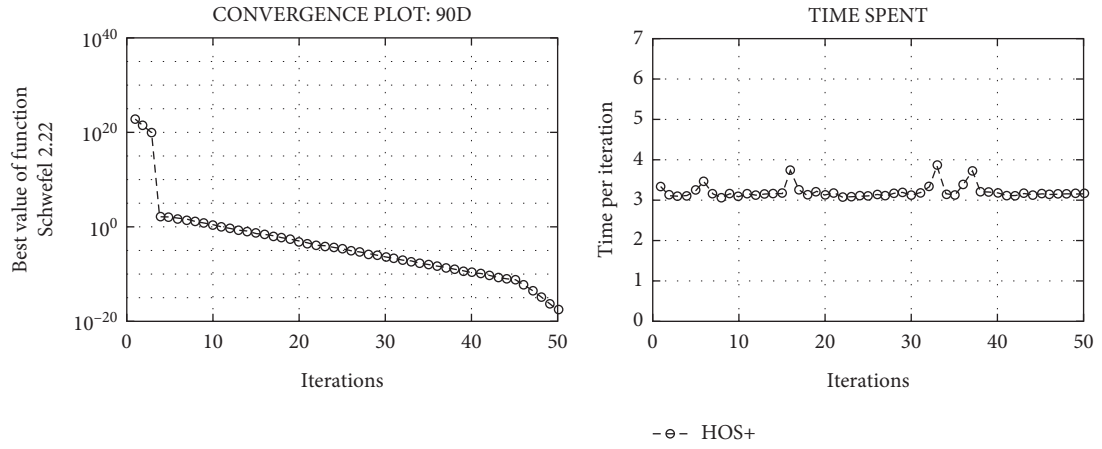


Figure 6: Convergence plot with time spent against 90-dimensional Schwefel 2.22 function (F2).

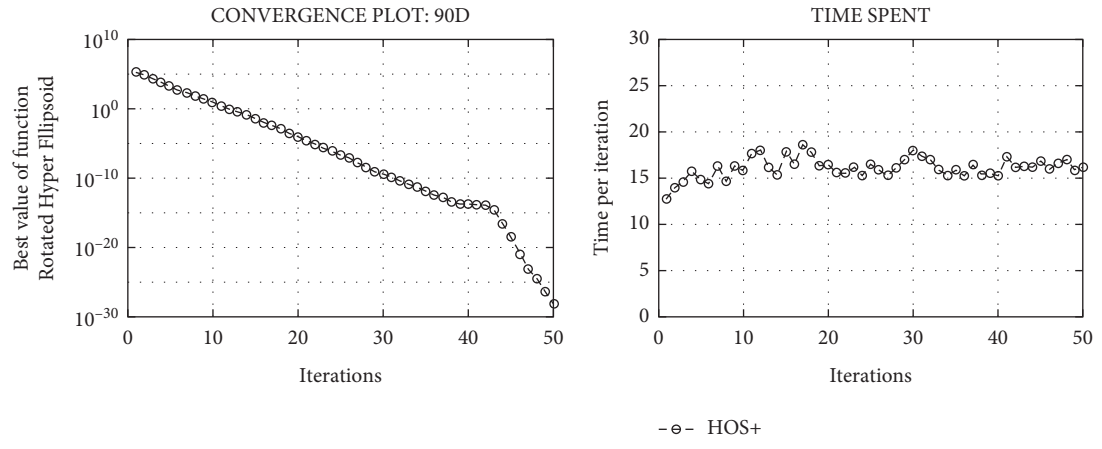


Figure 7: Convergence plot with time spent against 90-dimensional rotated Hyperellipsoid function (F3).

In the second stage, the HOS+ algorithm was compared with the meta-heuristic optimisation algorithms ACO, GA, DE, and PSO. Table 3 depicts the experimental results for each function.

In the third stage, the HOS+ algorithm was compared with the EHO, MSA, and WOA meta-heuristic optimisation algorithms. Table 4 depicts the experimental results obtained for each function.

In the fourth stage, the HOS+ algorithm was compared with the monarch butterfly optimisation algorithm (MBO), MBO with opposition-based learning and random local perturbation (OPMBO), and MBO with greedy

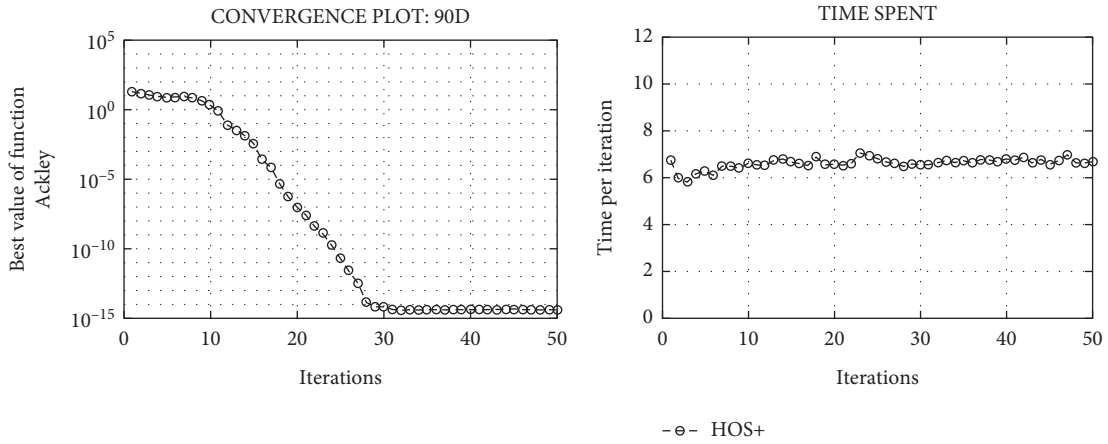


Figure 8: Convergence plot with time spent against 90-dimensional ackley function (F4).

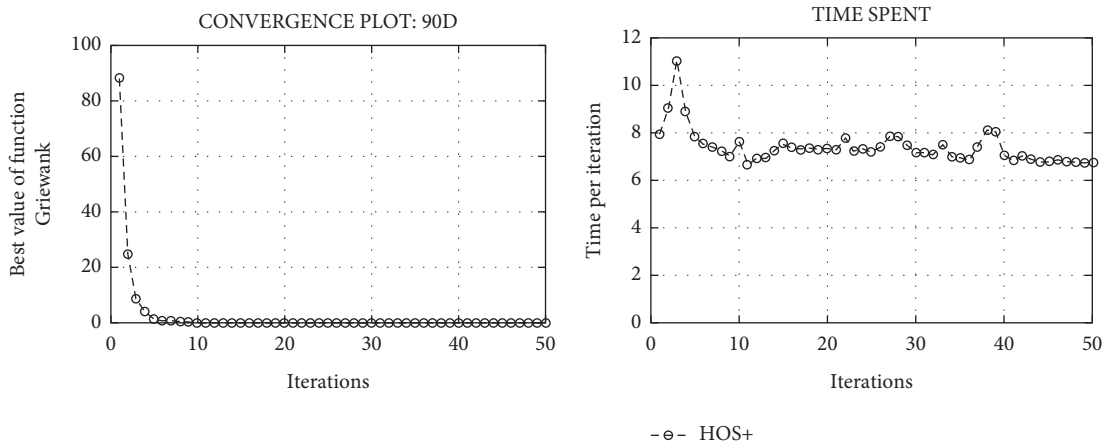


Figure 9: Convergence plot with time spent against 90-dimensional Griewank function (F5).

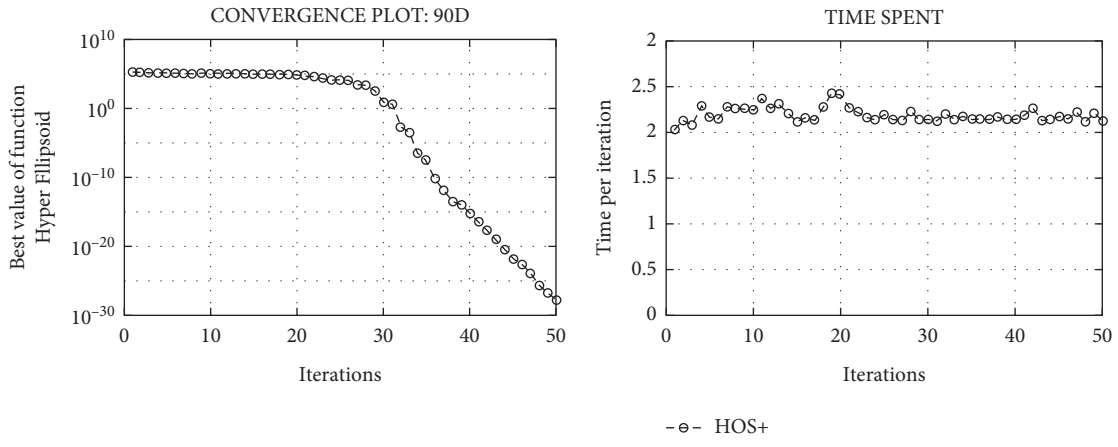


Figure 10: Convergence plot with time spent against 90-dimensional Hyperellipsoid function (F6).

strategy and self-adaptive crossover operator (GCMBO) using three benchmark functions. The comparative results are presented in Table 5. The best results are marked in bold.

In the fifth stage, the simulation results of HOS+ algorithm is compared with the simulation results of sine-cosine algorithm (SCA), m-SCA [34] and improved crow search algorithm ICSA [35]. Table 6 shows the experimental comparative results of the F1, F2, F3, F4 and F5 functions on 30 dimensions.

The experimental comparative results of ACO, ABC, DA DE, HAD, GA, PSO, EHO, MSA, WOA, MBO, GCMBO, OPMBO, ICSA, SCA, M-SCA and HOS+ algorithm showed

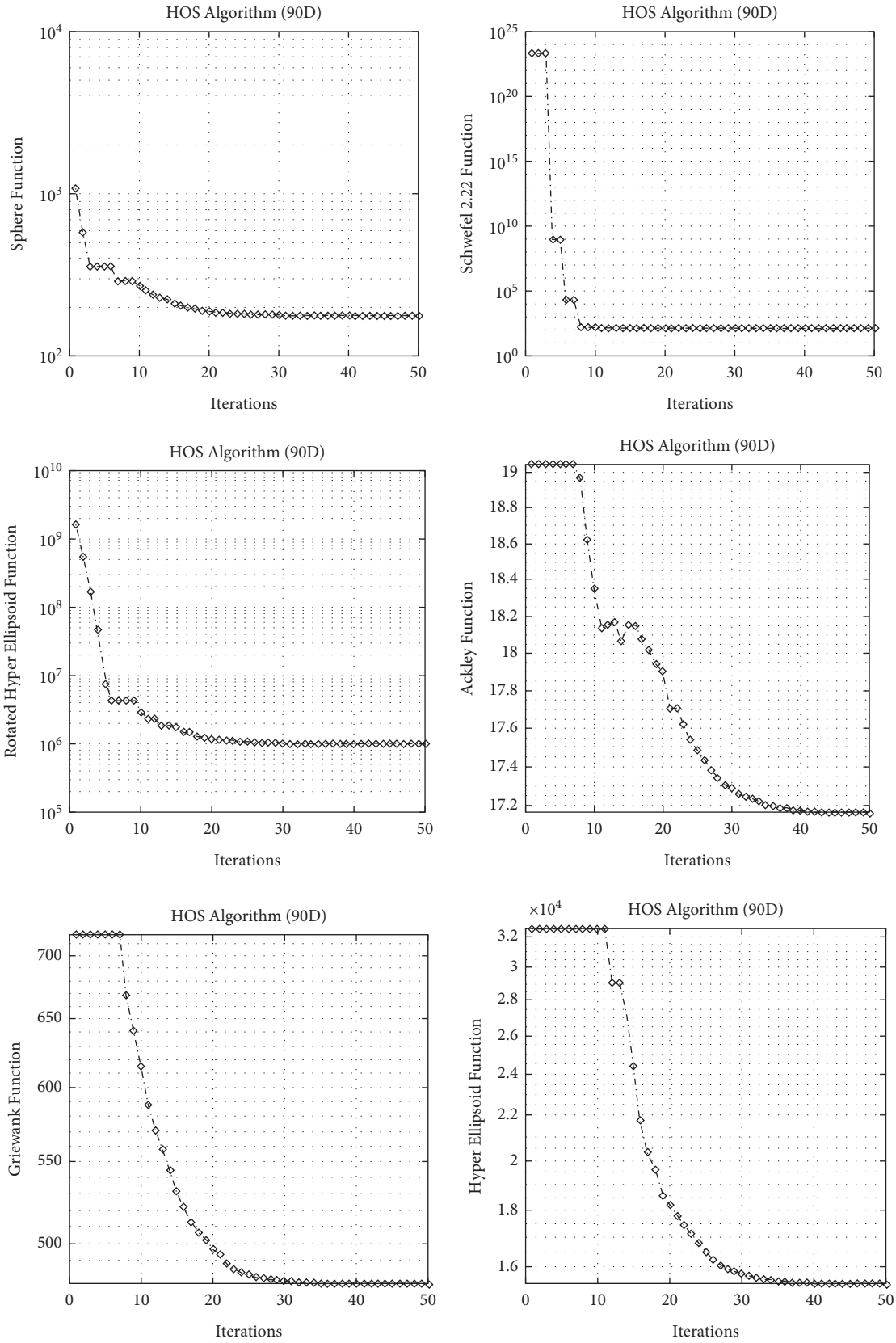


Figure 11: Convergence plots of original HOS algorithm for optimisation functions (F1–F6).

Table 2: The mean of test functions values found by ABC, DA, HAD, and HOS+

F	D	ABC	DA	HAD	HOS+
F1	30	$2.98E+01$	$1.76E+00$	$2.61E-14$	$2.63E-51$
	60	$1.97E+02$	$2.29E+00$	$3.46E-10$	$6.39E-39$
	90	$4.07E+02$	$8.53E+00$	$2.05E-09$	$2.19E-32$
F2	30	$7.54E+00$	$7.46E+00$	$3.40E-08$	$1.98E-27$
	60	$6.19E+01$	$2.80E+01$	$4.66E-06$	$3.88E-22$
	90	$1.59E+02$	$3.10E+01$	$4.89E-05$	$5.72E-18$
F3	30	$4.85E+04$	$1.96E+03$	$9.76E-14$	$2.69E-47$
	60	$9.03E+05$	$2.70E+04$	$2.56E-08$	$1.75E-34$
	90	$2.92E+06$	$1.53E+05$	$1.24E-07$	$1.35E-28$
F4	30	$1.70E+01$	$1.76E+00$	$5.89E-08$	$3.57E-15$
	60	$1.95E+01$	$7.41E+00$	$7.18E-06$	$2.53E-14$
	90	$2.00E+01$	$7.00E+00$	$3.08E-05$	$4.29E-13$
F5	30	$9.22E+01$	$7.64E+00$	$4.79E-12$	$0.00E+00$
	60	$6.93E+02$	$1.74E+01$	$8.68E-09$	$0.00E+00$
	90	$1.43E+03$	$2.27E+01$	$2.72E-07$	$0.00E+00$
F6	30	$2.36E+02$	$4.98E+00$	$5.16E-14$	$2.19E-48$
	60	$7.72E+03$	$1.41E+02$	$1.16E-10$	$3.39E-36$
	90	$3.85E+04$	$6.56E+02$	$2.09E-08$	$2.99E-29$

Table 3: The mean of test functions values found by ACO, DE, GA, PSO, and HOS+.

F	D	ACO	DE	GA	PSO	HOS+
F1	30	$1.63E+02$	$2.79E+01$	$9.58E+01$	$5.12E+01$	$2.63E-51$
	60	$3.76E+02$	$1.74E+02$	$2.86E+02$	$2.13E+02$	$6.39E-39$
	90	$6.02E+02$	$3.80E+02$	$4.65E+02$	$4.29E+02$	$2.19E-32$
F2	30	$1.13E+02$	$5.38E+01$	$8.60E+01$	$1.14E+02$	$1.98E-27$
	60	$2.48E+02$	$1.71E+02$	$2.03E+02$	$2.49E+02$	$3.88E-22$
	90	$3.88E+02$	$2.97E+02$	$3.23E+02$	$3.89E+02$	$5.72E-18$
F3	30	$2.13E+05$	$5.52E+04$	$1.41E+05$	$9.79E+04$	$2.69E-47$
	60	$1.65E+06$	$7.03E+05$	$1.18E+06$	$9.92E+05$	$1.75E-34$
	90	$4.30E+06$	$2.93E+06$	$3.20E+06$	$3.47E+06$	$1.35E-28$
F4	30	$1.85E+01$	$1.87E+01$	$1.77E+01$	$1.87E+01$	$3.57E-15$
	60	$1.90E+01$	$1.90E+01$	$1.86E+01$	$1.90E+01$	$2.53E-14$
	90	$1.91E+01$	$1.91E+01$	$1.88E+01$	$1.91E+01$	$4.29E-13$
F5	30	$8.57E+01$	$9.38E+01$	$1.27E+02$	$1.69E+02$	$0.00E+00$
	60	$4.32E+02$	$6.02E+02$	$4.64E+02$	$7.27E+02$	$0.00E+00$
	90	$7.13E+02$	$1.31E+03$	$8.87E+02$	$1.62E+03$	$0.00E+00$
F6	30	$2.34E+03$	$1.75E+02$	$1.20E+02$	$2.79E+02$	$2.19E-48$
	60	$2.28E+04$	$4.88E+03$	$7.93E+03$	$3.44E+03$	$3.39E-36$
	90	$8.42E+04$	$2.61E+04$	$4.28E+04$	$1.48E+04$	$2.99E-29$

Table 4: The mean of test functions values found by EHO, MSA, WOA, and HOS+.

F	D	EHO	MSA	WOA	HOS+
F1	30	$2.49E-07$	$2.30E-08$	$2.42E-09$	$2.63E-51$
	60	$6.44E-07$	$3.67E-07$	$4.67E-09$	$6.39E-39$
	90	$1.06E-06$	$1.17E-06$	$7.45E-09$	$2.19E-32$
F2	30	$4.12E-03$	$1.78E-04$	$3.76E-05$	$1.98E-27$
	60	$9.34E-03$	$9.50E-04$	$9.19E-05$	$3.88E-22$
	90	$1.46E-02$	$1.63E-03$	$1.52E-04$	$5.72E-18$
F3	30	$4.59E-04$	$3.02E-07$	$6.47E-06$	$2.69E-47$
	60	$2.41E-03$	$8.01E-06$	$2.97E-05$	$1.75E-34$
	90	$6.11E-03$	$4.79E-05$	$6.02E-05$	$1.35E-28$
F4	30	$1.94E-03$	$6.84E-05$	$1.21E-04$	$3.57E-15$
	60	$2.22E-03$	$1.88E-04$	$1.19E-04$	$2.53E-14$
	90	$2.33E-03$	$3.57E-04$	$1.41E-04$	$4.29E-13$

Table 4: Continued.

F	D	EHO	MSA	WOA	HOS+
F5	30	$1.44E-04$	$1.09E-09$	$6.10E-02$	$0.00E+00$
	60	$2.14E-04$	$7.15E-09$	$3.88E-02$	$0.00E+00$
	90	$2.58E-04$	$3.90E-08$	$3.47E-02$	$0.00E+00$
F6	30	$4.21E-06$	$3.33E-06$	$2.85E-08$	$2.19E-48$
	60	$4.57E-05$	$1.44E-04$	$3.31E-07$	$3.39E-36$
	90	$1.82E-04$	$9.07E-04$	$1.89E-06$	$2.99E-29$

Table 5: The mean of test functions values found by MBO, GCMBO, OPMBO, and HOS+.

F	D	MBO	GCMBO	OPMBO	HOS+
F1	20	$1.02E+01$	$4.03E-09$	$1.99E-10$	$4.69E-55$
	50	$2.09E+02$	$1.11E-09$	$8.85E-09$	$2.99E-44$
	100	$4.83E+02$	$1.00E+01$	$4.87E+00$	$1.20E-30$
F2	20	$1.95E+01$	$2.27E+00$	$1.72E-06$	$4.68E-33$
	50	$1.30E+02$	$2.36E+01$	$2.73E-05$	$1.69E-26$
	100	$2.66E+02$	$5.61E+01$	$7.60E-02$	$3.77E-16$
F4	20	$8.92E+00$	$2.61E-01$	$6.94E-06$	$1.29E-16$
	50	$1.59E+01$	$1.96E+00$	$4.03E-05$	$1.59E-15$
	100	$1.86E+01$	$8.63E+00$	$3.93E-02$	$3.69E-12$

Table 6: The mean of test functions values found by ICSA, SCA, M-SCA and HOS+ algorithms with $D=30$.

	F1	F2	F3	F4	F5
ICSA	$3.06E-5$	$8.15E-6$	$8.21E+2$	$2.80E-3$	$3.49E-2$
SCA	$1.87E+01$	$1.79E-02$	$8.84E+03$	$1.83E+01$	$9.38E-01$
M-SCA	$5.70E-03$	$9.11E-04$	$8.48E+02$	$3.36E-03$	$3.84E-02$
HOS+	$2.63E-51$	$1.98E-27$	$2.69E-47$	$3.57E-15$	$0.00E+00$

that the proposed HOS+ algorithm has obtained better results and best convergence due to escaping local optimums in the majority of the evaluations. The obtained simulation results indicate the effectiveness of using HOS+ algorithm in optimisation problems.

6. Conclusions

This paper proposes a novel stochastic search algorithm based on the evolution of hypercube. The design stages of the algorithm were explained. A new random perturbation module is introduced in order to solve local optimum problems in optimisation problems and to find a global solution. The HOS+ algorithm has been tested using various low and high dimensional optimisation functions and the solution of the specified local optimum problem has been proven by experimental results. The obtained results demonstrated that the algorithm can successfully avoid getting stuck in the local optimum and find a global solution for the lowest iterations. Comparative results of performances that include the best, the mean, standard deviation and convergence plots demonstrate advantages of the proposed HOS+ algorithm over other thirteen meta-heuristic algorithms. The obtained simulation results indicate the efficiency of using the HOS+ algorithm in the solution of optimisation problems. Future research includes the application of the HOS+ algorithm to solve practical optimisation problems.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] R. H. Abiyev and M. Tunay, "Optimization of high dimensional functions through hypercube evaluation," *Computational Intelligence and Neuroscience*, vol. 2015, Article ID 967320, 11 pages, 2015.

[2] R. H. Abiyev and M. Tunay, "Optimization search using hypercubes," in *Proceedings of the 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, pp. 1–8, Istanbul, Turkey, October 2020.

[3] S. Srivastava and S. K. Sahana, "Application of bat algorithm for transport network design problem," *Applied Computational Intelligence and Soft Computing*, vol. 2019, no. 5, pp. 1–12, Article ID 9864090, 2019.

[4] M. Mareli and B. Twala, "An adaptive Cuckoo search algorithm for optimisation," *Applied Computing and Informatics*, vol. 14, no. 2, pp. 107–115, 2018.

[5] S. Mirjalili, "The ant lion optimizer," *Advances in Engineering Software*, vol. 83, pp. 80–98, 2015.

[6] K. Roy, K. K. Mandal, and A. C. Mandal, "Ant-Lion Optimizer algorithm and recurrent neural network for energy

management of micro grid connected system," *Energy*, vol. 167, pp. 402–416, 2019.

[7] G.-G. Wang, S. Deb, and L. S. Coelho, "Elephant herding optimization," in *Proceedings of the 3rd International Symposium on Computational and Business Intelligence (ISCBI)*, pp. 1–5, Bali, Indonesia, December 2015.

[8] J. Li, H. Lei, A. H. Alavi, and G.-G. Wang, "Elephant herding optimization: variants, hybrids, and applications," *Mathematics*, vol. 8, no. 9, 2020.

[9] Y. Li, X. Zhu, and J. Liu, "An improved moth-flame optimization algorithm for engineering problems," *Symmetry*, vol. 12, no. 8, 2020.

[10] G.-G. Wang and A. H. Gandomi, "A comprehensive review of krill herd algorithm: variants, hybrids and applications," *Artificial Intelligence Review*, vol. 51, pp. 119–148, 2019.

[11] I. Strumberger and N. Bacanin, "Modified moth search algorithm for global optimization problems," *International Journal of Computer*, vol. 3, pp. 44–48, 2018.

[12] H. Hu, Z. Cai, S. Hu, Y. Cai, J. Chen, and S. Huang, "Improving monarch butterfly optimization algorithm with self-adaptive population," *Algorithms*, vol. 11, no. 5, 2018.

[13] Y. Feng, S. Deb, G. G. Wang, and A. H. Alavi, "Monarch butterfly optimization: a comprehensive review," *Expert Systems with Applications*, vol. 168, Article ID 114418, 2021.

[14] A. Abusnaina, R. Abdullah, and A. Kattan, "Self-adaptive mussels wandering optimization algorithm with application for artificial neural network training," *Journal of Intelligent Systems*, vol. 29, no. 1, pp. 345–363, 2020.

[15] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016.

[16] M. Georgioudakis and V. A. Plevris, "Comparative study of differential evolution variants in constrained structural optimization," *Frontiers in Built Environment*, vol. 6, 2020.

[17] U. Guvenc, A. Isık, T. Yigit, and I. Akkaya, "Performance analysis of biogeography-based optimization for automatic voltage regulator system," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 24, no. 3, pp. 1150–1162, 2016.

[18] H. Garg, "An efficient biogeography based optimization algorithm for solving reliability optimization problems," *Swarm and Evolutionary Computation*, vol. 4, pp. 1–10, 2015.

[19] X. Z. Gao, V. Govindasamy, H. Xu, X. Wang, and K. Zenger, "Harmony search method: theory and applications," *Computational Intelligence and Neuroscience*, vol. 2015, Article ID 258491, 10 pages, 2015.

[20] Z. Li, X. Lin, Q. Zhang, and H. Liu, "Evolution strategies for continuous optimization: a survey of the state-of-the-art," *Swarm and Evolutionary Computation*, vol. 56, Article ID 100694, 2020.

[21] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," *Knowledge-Based Systems*, vol. 96, pp. 120–133, 2016.

[22] E. Rashedi, E. Rashedi, and H. Nezamabadi-pour, "A comprehensive survey on gravitational search algorithm," *Swarm and Evolutionary Computation*, vol. 41, pp. 141–158, 2018.

[23] R. H. Abiyev and M. Tunay, "Experimental study of specific benchmarking functions for modified monkey algorithm," *Procedia Computer Science*, vol. 102, pp. 595–602, 2016.

[24] M. Tunay, "A new design of metaheuristic search called improved monkey algorithm based on random perturbation for optimization problems," *Scientific Programming*, vol. 2021, Article ID 5557259, 14 pages, 2021.

[25] C. Rahman and T. Rashid, "A survey on dragonfly algorithm and its applications in engineering," *Journal of Computational Design and Engineering*, vol. 2020, pp. 1–23, 2020.

[26] W. A. H. M. Ghanem and A. Jantan, "A cognitively inspired hybridization of artificial bee colony and dragonfly algorithms for training multi-layer perceptrons," *Cognitive Computation*, vol. 10, pp. 1096–1134, 2018.

[27] G.-G. Wang, S. Deb, and Z. Cui, "Monarch butterfly optimization," *Neural Computing & Applications*, vol. 31, pp. 1–20, 2015.

[28] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: a new method for stochastic optimization," *Future Generation Computer Systems*, vol. 111, pp. 300–323, 2020.

[29] G.-G. Wang, "Moth search algorithm: a bioinspired metaheuristic algorithm for global optimization problems," *Memetic Computing*, vol. 10, pp. 151–164, 2016.

[30] Y. Yang, H. Chen, A. A. Heidari, and A. HGandomi, "Hunger games search: visions, conception, implementation, deep analysis, perspectives, and towards performance shifts," *Expert Systems with Applications*, vol. 177, Article ID 114864, 2021.

[31] J. Tu, H. Chen, M. Wang, and A. H. Gandomi, "The colony predation algorithm," *J Bionic Eng*, vol. 18, pp. 674–710, 2021.

[32] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: algorithm and applications," *Future Generation Computer Systems*, vol. 97, pp. 849–872, 2019.

[33] L. Sun, S. Chen, J. Xu, and Y. Tian, "Improved monarch butterfly optimization algorithm based on opposition-based learning and random local perturbation," *Complexity*, vol. 2019, Article ID 4182148, 20 pages, 2019.

[34] S. Gupta and K. Deep, "A hybrid self-adaptive sine cosine algorithm with opposition based learning," *Expert Systems with Applications*, vol. 119, pp. 210–230, 2018.

[35] M. J. Aliabadi and M. Radmehr, "Optimization of hybrid renewable energy system in radial distribution networks considering uncertainty using meta-heuristic crow search algorithm," *Applied Soft Computing*, vol. 107, Article ID 10738, 2021.

Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 2678195, 11 pages
https://doi.org/10.1155/2022/2678195

Research Article

Control and Implementation of Positioning System with Symmetrical Topology for Precision Manufacturing

Quang Vinh Truong[1,2] **and Ha Quang Thinh Ngo**[2,3]

[1]*Department of Electronics, Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Vietnam*
[2]*Vietnam National University Ho Chi Minh City (VNU-HCM), Linh Trung Ward, Thu Duc District, Ho Chi Minh City, Vietnam*
[3]*Department of Mechatronics, Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Vietnam*

Correspondence should be addressed to Ha Quang Thinh Ngo; nhqthinh@hcmut.edu.vn

Received 8 September 2021; Revised 12 October 2021; Accepted 21 October 2021; Published 25 February 2022

Academic Editor: Guoqiang Wang

In recent years, the needs of modularized controller for the multiaxes servo system increase significantly since traditional controller still exists many drawbacks such as limited control axes, low speed data acquisition, or heavy weight. In this paper, we present the design and implementation of both hardware and software for real-time express-based motion controller. This controller can meet the demand for high-speed motion control and high performance which conventional fieldbus controllers cannot realize. With modular design, the controller brings many benefits such as low-cost, expandable ability, multiaxes control, or small physical size. Experimental results for an industrial motion system indicate that the proposed modular controller can perform well in time critical data transmission and is feasible and applicable in various fields.

1. Introduction

Motion control system, which plays as an important role in the modern industry such as Computed Numerically Controlled (CNC) machine tools [1–4] and industrial robots, determines the whole system performance. For the demand for high reliability and high real-time manufacturing, modern motion control systems are growing in the direction of high speed, high precision, multifunctional, and opening that needs embedded and networked technique to implement. Networked motion control systems are special networked control systems [5–8] composed of controllers and multiaxes motor drivers, which can ensure the real-time synchronization of motion control order and motion state transmission.

A lot of buses based on Ethernet have been introduced into the motion control field, such as PROFIBUS, CAN bus, Interbus-S, and SERCOS [9–12]. C. C. Hsieh and P. L. Hsu [13, 14] proposed a CAN-based motion control system by introducing the event-time structure. Thus, the CAN-based motion control system achieves desired control performance even under lower transmission rate or heavier message loading on the bus. B. Chen et al. [15, 16] introduced three configurations of the networked synchronized motion control systems. In a networked synchronized motion control system, short communication cycle is needed in order to upgrade the synchronization accuracy. In other approaches, the authors are successful in developing the motion control system with the modern industry Ethernet buses. Y. Hu et al. [17, 18] reconfigured the open architecture CNC system based on the Mechatrolink-II Fieldbus [19, 20] that offers interoperability and portability. This component-based CNC architecture is flexible and reconfigurable and has more precise performance.

In time critical environment or where complex multitasking operations are employed, the execution time of such software can prove a significant overhead on the system. The development of a real-time executive consisting of both

hardware and software codesign methodologies was introduced firstly [21, 22]. The researchers utilized this real-time executive as a prototyping tool for investigating different task scheduling schemes based on high performance evolutionary heuristics. Some of the results were obtained by implementing a Hopfield neural network to solve the task scheduling problem in embedded and real-time systems. However, this approach may be not expandable in the large-scale system, and it depends on each application. Related to motion control, the embedded framework and network communication are the cores of multiaxes motion controller. In [23–26], one kind of modular control component was presented. The network protocol from servo A6N could provide a huge chance to achieve high performance and advanced motion function. Nevertheless, it lacks graphical user interface to assist operator, limited function, and big size.

2. Related Works and Problem Statement

2.1. Related Works. The most common implementation of a control system in any process industry is through a central controller. A single controller is directly attached with its input and output devices by cables in a point-to-point mode. When hundreds of field devices are employed, enormous lengths of special cables are used. It led to the occurrence of a novel concept as fieldbus which is a digital, bi-directional, serial-bus communications network supposed to link various instruments, transducers, controllers, final control elements, process stream analyzers, and computer control systems. Nowadays, the considerable developments in fieldbus protocols in Table 1 have launched many modern industrial networks such as CC-Link and SSCNET from Mitsubishi, Profibus from Siemens, or DeviceNet from Allen-Bradley. In recent years, several advanced network protocols are introduced, for instance, RTEX from Panasonic or EtherCAT from Beckhoff.

Communication network protocol is one of the industrial networks based on RS485 utilizing the protocol standard of CC-Link [27]. Generally, the working mode on CC-Link consists of cyclic transmission and instantaneous transmission in which broadcast polling mode is mainly integrated. At initial stage, the master station firstly sends the acquired data to all slave stations. The data of synchronization mode on broadcast polling are performed by frame synchronization, and the instantaneous transmission must be completed by sending traffic. The automation of industrial production can be improved that CC-Link protocol well-established the communication between Programmable Logic Controller (PLC) and slave stations. However, in several applications which require the interactions with human, CC-Link does not match since the intelligent machine becomes the inevitable trend in our era. Therefore, a PC-based application in multiaxes synchronous motion such SSCNET-based system [28] could satisfy the modern requirements. In reality, computer-aided manufacturing has been no stranger anymore. The parametric curve interpolators attracted many researchers on investigating the analytical shapes, values of control points, weights, and knots. As the complexity of mathematical expression, the transmitting time could be costed so much. Hopefully, based on the powerful computer, the real-time performance of this system could be maintained.

In the context of competitive responses, it is crucial to evaluate the technical specifications of the motion networks. For instance, a study on comparison of Profinet and Profibus DP [29, 30] has been presented in order to provide a systematic view of communication protocol. The data collection is the input resource to estimate some key indicators. The experimental tests consist of a frequency converter, an electric AC motor, and an incremental encoder. Under these circumstances, the dynamics in the motion control network are to send the reference position by the controller via communication protocol to frequency converter. Then, the network data flow in various cycle times is gathered by a measurement system during a certain period while application performs position control. For the specialized controller network, CANopen is a real-time protocol for the distributed control system [31, 32]. To enhance interoperability and interchangeability among different products from dissimilar manufacturers, practical applications usually make use of communication services defined at the application layer, which are significantly more flexible than those provided originally by the data-link layer.

In these days, the advanced fieldbus technologies are presented by RTEX or EtherCAT. Both of them which derived from the Ethernet protocol contain the fundamental layers of general data frame. They also have high performance and flexible topology and deliver more options to user. In some reports [33, 34], EtherCAT is an overall highly performance real-time protocol, for example, high velocity of transmission, multislave nodes, various suppliers and manufacturers, and both academic and industry community. Contrary to its advantages, the existing drawbacks which comprise limited usages in software and hardware, or complex data structure, need more solutions to overcome. Many investigators in Panasonic Corporation [35] have innovated the real-time protocol for a high-speed synchronous motion network, named as RTEX. This system achieves fully synchronization, low-cost device, simple structure, and noise rejection. Industries like semiconductors, liquid crystal manufacturing equipment, and electronic parts mounter are strongly recommended to adopt this technique.

2.2. Problem Statement. Recently, there exists an innovative trend in multiaxes control that enables the fast response, supports the large number of slave devices, configures less wiring of mechanism, and ensures the high reliability. It is hard for previous network systems to be quite adaptive, but these constraints explicitly fit in well with real-time express protocol. This standard communication is primarily introduced by Panasonic group, and up-to-now it is one of the well-known network motion system in the global market. Hereafter, the contributions of this paper are that (i) a novel design of network motion module has been developed successfully. Adopting the framework of the embedded

Table 1: Summary of comparative reviews.

Authors	Methodology	Advantage(s)	Limitation(s)
[27] Yeon and Kim	A new CC-Link module utilizing R-IN32M3 to improve the expandability is proposed.	The probabilities of data loss could be managed and interference with equipment control could be decreased.	It requires a lot of pins because of parallel communication.
[28] Cheng and Tseng	Since the cutter contact (CC) velocity along the surface tends to vary, a PC-based real-time motion control network was investigated to achieve the goal of multiaxes synchronous motion.	Both the NURBS interpolator algorithm and SSCNET protocol are successfully realized with the RTX real-time kernel.	The burden computational cost on Windows-OS might cause a problem in traffic communication. Only limited axes are validated.
[29] Dias et al. and [30] Khaliq et al.	This work analyses the performance of industrial communication network by comparing Profinet and Profibus DP in cycle time and jitter.	It is found that Profibus DPV0 is faster and Profinet version IRT has higher determinism.	It does not allow to set fixed cycle time when working. These technologies are not newly developed and not much user utilized them.
[31] Cena and Valenzano, and [32] Liu et al.	A new automatic node discovery protocol for unregistered nodes when connected to the network is investigated.	This research helps to identify the nonconfigured nodes directly and provides a very good degree of compatibility with the existing devices.	Each device is assigned a unique node address and does not permit to alter online.
[33] Chen et al. and [34] Langlois et al.	The system structure evolves one master and slaves connected into a loop. Master device employs 100BASE-TX Ethernet adapter while slave controller utilizes an ASIC or a FPGA with EtherCAT IP core to handle the data frame.	Rapid transmission, numerous slave nodes, diverse manufacturers, and large academic community are benefits of EtherCAT.	It involves the licensed software, genuine equipment to gain the highly precise performance, and complicated data model to understand.

system, it was integrated into new network communication with highly real-time performance and stable protocol. In this current trend, the control issue of multiaxes becomes more critical, especially in the large-scale system. (ii) The problems that were addressed in synchronous control, network topology, time consuming, or maintenance service have been solved this by the real-time express method. (iii) The experiments with the practical servo system are established in order to prove the validity of our approach and the ability of powerful network control. The rest of this paper is organized as follows. Section 3 introduces some definitions, technical specifications, and analysis of the real-time system. In Section 4, the detailed development of hardware platform and several notifiable remarks are carried out. Also, this section briefly describes the control software which is programmed in C++ and interacted during a cycle servo. The results of experimental module have been completed to validate our proposed approach in Section 5. Several discussions consisting of experiences, practical performance, or tested method are revealed together with the conclusions with future work in Section 6.

3. Proposed Approach of Network Control

3.1. Introduction. Real-time express is a high-speed synchronous motion network developed by Panasonic Corporation. While being 100 Mbps very high speed (ten times higher than our previous model), the system cost can be kept low by using the commercially available LAN cable. It supports fully synchronization, full-duplex communication, and noise immunity with unique error correction. This protocol is connected via ring topology as shown in Figure 1 with advantages of simple data flow effects, high efficiency, and reliability. Comparing with line topology transferring data through many nodes which causes low efficiency, there is no cross-talk in ring structure that occasionally is sources of troubles in high-speed data. With only one master and many slaves, a large-scale system could be monitored and managed easily.

3.2. Description of New Concept. MNM1221 is a serial interface controller ASIC that enables to establish the real-time communication systems based upon the master-slaves communication. The MNM1221 requires to be used with a PHY (physical layer chip), a pulse transformer, and shielded twisted pair cables for 100BASE-TX (IEEE 802.3u) system. In other words, MNM1221 is a special MAC (Media Access Controller) in order to suit 100BASE-TX to the real-time communication system for the multiaxes servo control.

The MNM1221 serial interface system consists of one master and several slaves and exchanges the command data from the master and the response data from the slave cyclically. For that, MNM1221 has double banks (buffer) memory for each transmitting and receiving, and this function allows the CPU to operate efficiently.

Table 2 illustrates the information of period, axes, and modes. The number of controlled axes depends on different sources such as transferring data mode, communication period, and update period. The timing communication is a period of frame transmission while update cycle is to inform data inside the frame. The external device should be limited

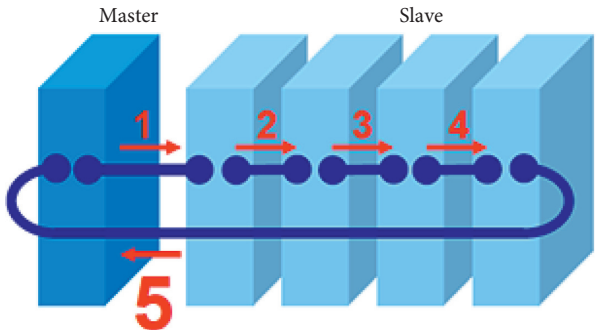


Figure 1: Structure specification of ring topology.

Table 2: Combination of period, axes, and modes.

Update period (ms)	Com. period (ms)	Max of axes
4	2	32
2	2	32
2	1	32
1	1	32
1	0.5	32
0.5	0.5	32
0.5	0.25	16
0.25	0.25	16
0.25	0.125	8
0.125	0.125	8
0.125	0.0625	4

to join in network since it impacts on number of controlled axes.

4. Design of Real-Time Express Module

4.1. Data Exchange. Traditional Ethernet is not proper to satisfy real-time performance of the motion system because of time delay, synchronous transmission, or data frame format. To overcome these drawbacks, a module of real-time express communication which is already well-defined to Industrial Ethernet solutions is proposed. Figure 2 represents the whole linkage in master unit. Once, the operator manipulates on host personal computer in central control room. The popular USB connection assists easy plug-in and fast data transportation between host and network module. There are two selections in circuit schematic, debug for development and release for commerce. The host controller links with servo pack by two LAN wires (receive and transmit) which could extend up to 100 meters. The A6N servo pack-based system is our target of motion controller. To visualize the feed-back signals, a built-in software installed in host computer interconnects with servo via USB type-B.

For the internal bus, the inside hardware schematic is designed in Figure 3. To expand more controlled axes, 16-byte mode is chosen by connecting BUSMODE pin to ground. The address bus access is from 0×000 h to 0×200 h for transmit memory, from 0×200 h to 0×400 h for receive memory, from 0×400 h to 0×480 h for control registers, and the rest for status registers. The timing schedule could be guaranteed owing to CS, RD, WR, and WAIT pins. The real-time express protocol IC has two timer sources that are an

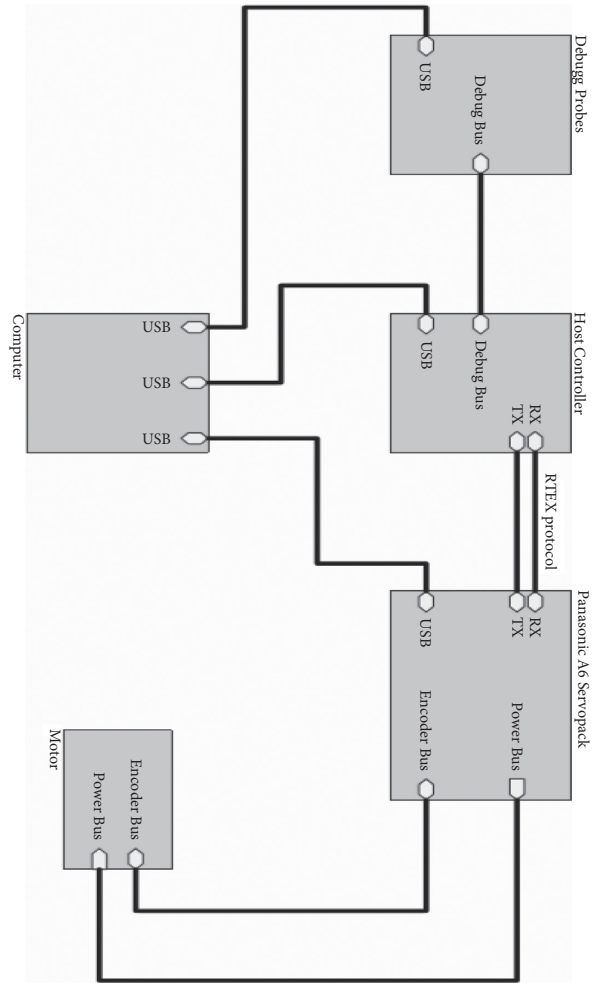


Figure 2: Block diagram of the whole master module.

external timer and an internal timer as shown in Figure 4. For external timer design, it requires that microprocessor provides a tick-clock signal to coordinate the operation of transmit and receive. In the case that using internal timer, MNM1221 utilizes its timer independently. The physical layer chip needs to configure the suitable operating mode from network protocol IC.

Communication period, that is, the same as transmitting period, cannot be set up freely because it must be synchronized with servo control. According to servo specifications, the period must be set to suitable value as accurately as possible. In servo A6N, the communication period must be selected from 2 ms, 1 ms, 0.5 ms, 0.25 ms, 0.125 ms, and 0.0625 ms. The command update period is the same or twice as the communication period and must be selected from 4 ms, 2 ms, 1 ms, 0.5 ms, 0.25 ms, and 0.125 ms. The default setting of the servo is that the communication period is 0.5 ms and the command update period is 1 ms.

4.2. Design of Control Exchange. In the firmware level, most of operations have been handled by CPU as shown in Figure 5. Initially, all variables in program should be given default

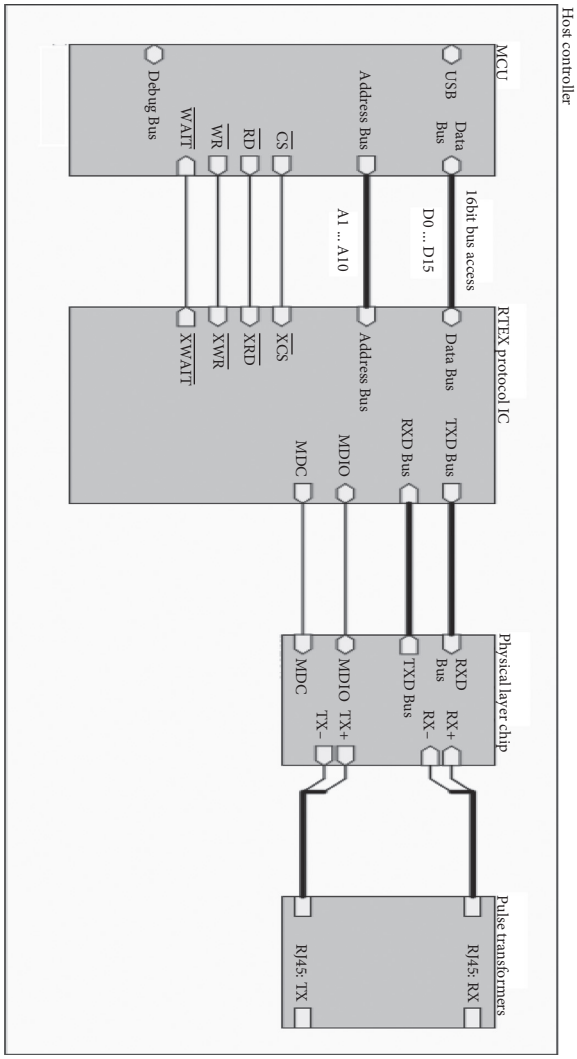


Figure 3: Hardware connection among microprocessor unit, real-time express ASIC chip, and peripheral devices.

values. Then, several registers of real-time express IC need to set up operating mode such as control registers (M_INIT_-DONE, M_INITF_TX, M_TXTIM_SEL) and status registers (M_STATE, M_NODE_SUM). The communication in network could be enabled by checking registers (M_TXMEM_SW, M_CYCL_START). There is an infinite loop which often reads data from status registers. As a result, the data from numerous slaves are updated frequently.

Since the synchronous interface plays an important role, a routine to incorporate among them is displayed in Figure 6. When slave sends a request to master, if this message is firstly notified, the readout pointer would record it. To verify the notification message, the process of redundancy check is cyclically executed. If there exists any error in transmission, the counter would label to monitor internal status. Later, this error count is stored in master station. In the case that there is no error, the receive buffer updates from network motion IC to internal side. Maybe, user's application takes this information, performs the execution, and releases currently internal state to network.

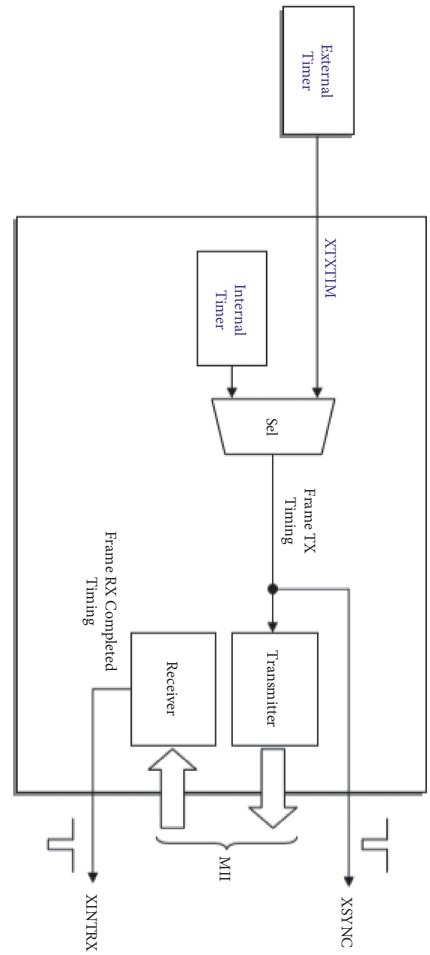


Figure 4: Timer sources of timing frame data.

5. Results and Discussion

5.1. Laboratory Experiment. To verify the feasible, capable, and applicable design in our approach, an experimental test is carried out as shown in Figure 7. The host PC is Dell Latitude 5500 with powerful Intel core processor. Most of control software would execute in Windows 10 operating system as shown in Figure7(a). The software-based work is mainly programmed by Visual Studio with various C++ classes supported from Windows. The firmware which is written by C language handles the important role in data processing and exchanging. The information is traded between host and motion module through USB cable. Two LAN wires (from RX port of module to TX port of A6N servo pack and vice versa). One remarkable point is that this module offers on-the-fly adjustment parameters. Hence, as soon as plugging into port, the data flow has been established continuously. Furthermore, the integrated emulator should be utilized to track each missing command or system error as shown in Figure 7(b). Its benefit is to shorten the debugging period which is usually longer.

The laboratory validation on one axis is shown in Figure 8. In S-curve motion profile which is default mode, the command code (0×10) is transmitted from master to servo pack. Together with its, target position and target speed are

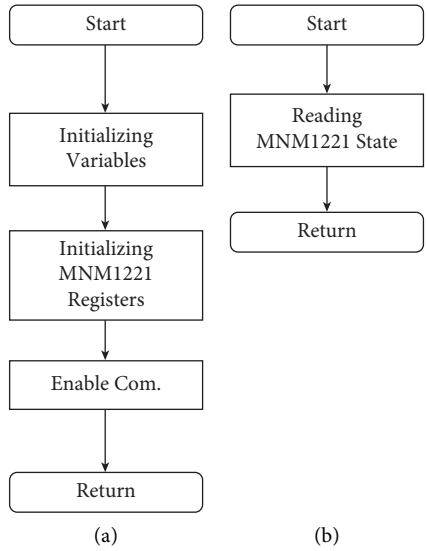


Figure 5: Flowchart of control firmware: (a) initializing and (b) main loop.

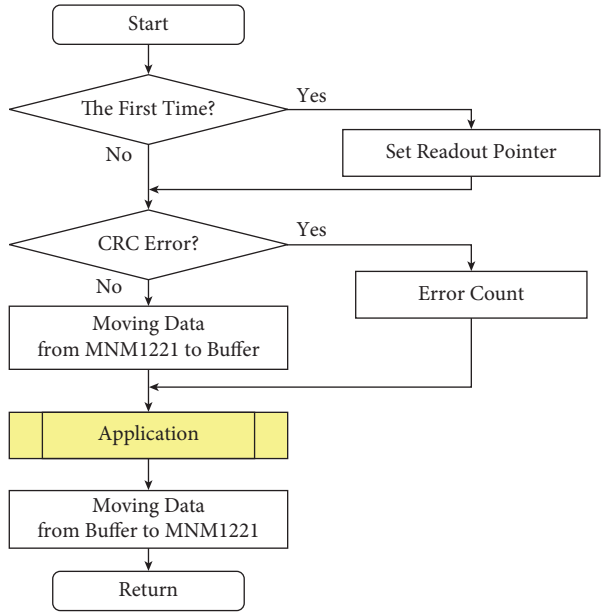
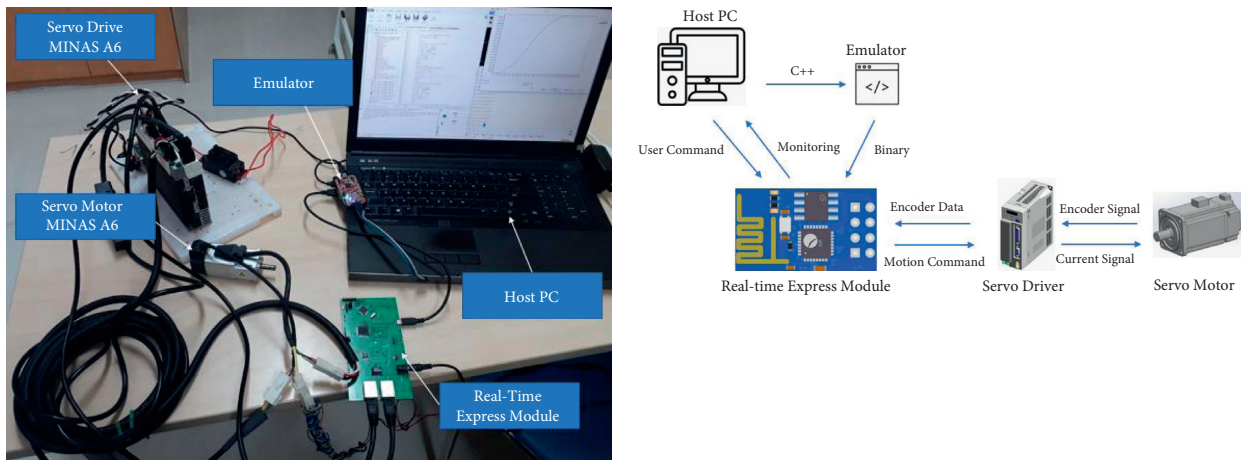


Figure 6: Flowchart of synchronizing routine.

written in data frame. Optionally, stop mode in byte 3 and absolute/incremental mode in byte 8 are set at start. Then, data frame is shifted to each node in the ring topology. The experimental results of symmetrical profile for position, velocity, acceleration, and jerk are illustrated in Figures 8(a)–8(d), respectively. During one cycle, a master module collects responses from all slaves. In the feed-back frame, except practical data, the servo status consisting of servo ready, alarm, warning message, or in position signal aids master to monitor online.

For asymmetric S-curve profile from Figures 8(e) to 8(h), it involves more setting parameters before executing. The values of acceleration and deceleration are registered in parameter 8.01 and 8.04. Several secondary parameters, for instance, digital filter, could bring more clear and enhanced performance. Figures 8(e)–8(h) present actual position, velocity, acceleration, and jerk in A6N servo pack. The technical specifications of both profiles are listed in Table 3. Generally, the symmetric motion profile must ensure the same period for acceleration and deceleration while the asymmetric one generates more time sequences to decelerate the servo motor. Hence, the duration of S-curve is shorter than AS-curve although the similar inputs are supplied. Actual position of AS-curve profile is slightly better because it slowly tends to target location. Some advantages of the asymmetrical profile are to suppress the oscillation-induced motion and flexibly generate the user-defined profile. The accuracy control in max velocity and max acceleration of



(a) (b)

Figure 7: Experimental setup with proposed network module (a) and graphical presentation of the whole system (b).

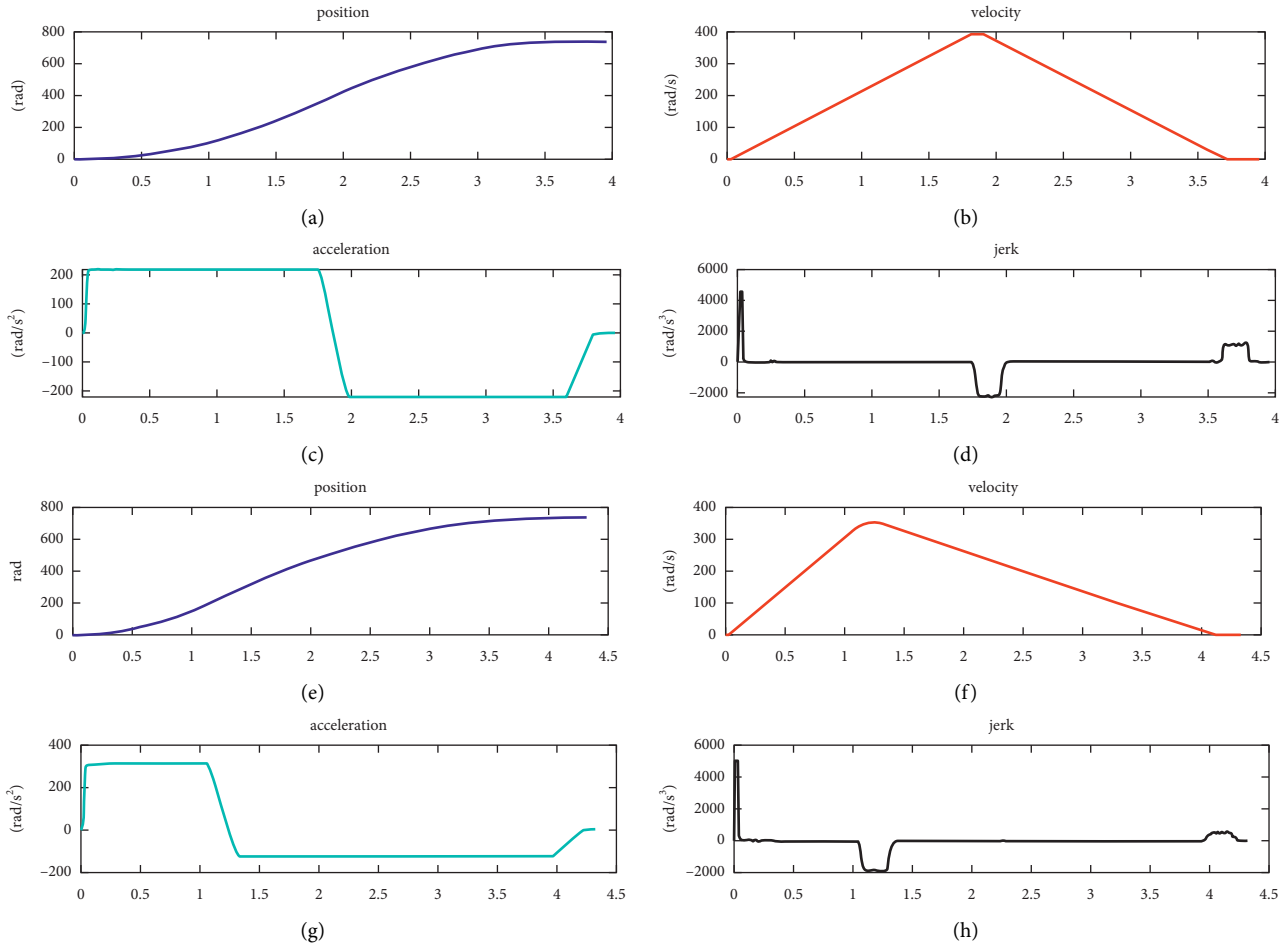


Figure 8: Experimental results of monoaxis control with S-curve profile (left: (a) position, (b) velocity, (c) acceleration, and (d) jerk) and AS-curve profile (right: (e) position, (f) velocity, (g) acceleration, and (h) jerk).

symmetric motion profile could be beneficially achieved since it does not need to decrease so much. At initial stage of asymmetric profile, it must peak as soon as possible, and then it generates the deceleration trajectory. As the abovementioned analysis, the advantageous specifications of asymmetrical S-curve are higher than those of symmetrical one. In the proposed design, the modular controller supports both of them with competitive performance.

Table 3: Descriptions of motion control parameters.

Motion profile	Duration	Command position (rad)	Command velocity (rad/s)	Command acceleration (rad/s^2)	Command jerk (rad/s^3)	Actual position (rad)	Actual velocity (rad/s)	Actual acceleration (rad/s^2)	Actual jerk (rad/s^3)
Symmetric	3.8 s	780	400	200	5000	781	397	200	4596
Asymmetric	4.25 s	780	400	200	5000	780	371	314	5082

5.2. Discussion. The communication cycle significantly impacts on overall performance of the network motion system. It should be considered carefully when designing hardware schematic. This section would analyze and discuss about influence of servo cycle and developing experiences.

To control timing in network, the XINTRX, XSYNC, and XTXTIM pins handle separated tasks. The XINTRX signal would activate when the procedure of receiving frame is completed. If using an external timing signal to start transmitting the frame in the running state, the cyclical trigger signal must send the acknowledgment signal to XTXTIM pin. In this case, the register setting must be needed to switch from the internal timing signal. Reversely, the XSYNC signal is changed logic level when the procedure of transmitting frame starts. In other words, after all slaves receive the data frame in running state, this pin outputs signal in all slaves to synchronize timing pulse. Therefore, it must be ensured that XSYNCs of all slaves are output at the same time as shown in Figure 9.

General speaking, there are four main states in real-time express protocol IC. The initial state starts firstly to reset registers. In the ring config state, network ASIC chip searches for the information of each slave and configures the operation. The result of this period is to store data into the status registers. After checking the correct slave information by comparing with sampled data, the firmware instructs this chip from ready state to proceed to running state. Otherwise, it returns an error message to notify there is no matched data in network. In running state, using TX and RX memory bank, the firmware is communicated cyclically. The transmission buffer memory (TX memory) and reception buffer memory (RX memory) are composed of two banks, respectively. To prevent conflict of data access, one bank is dedicated to the external CPU and another bank is dedicated to the internal communication module. Alternately, the assignment of two banks is switched before reading the received data.

The timing communication during four states is established in Figure 10. To implement rapidly, the duration of internal timer is equalized to communication period. This timer would provide signal to XTXTIM and guarantee that the low or high width of pulse must keep minimum 1 us. In ring config state, a network scan to diagnose slaves which presented in topology should not exceed a period of 2 ms. Once, entering the ready phase, the data transmission has already validated correctly and the registered slave information in master match with up-to-date ones. The transmission speed in running phase is very fast and repeats every cycle. Its speed depends on user's applications and hardware limitation.

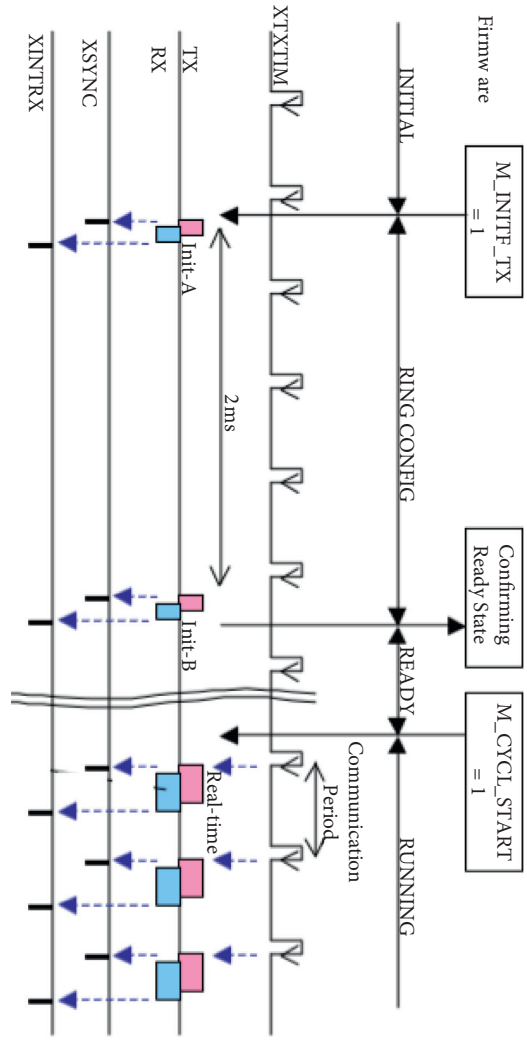


Figure 9: Timing control procedure in transmission protocol.

Furthermore, we verify the effectiveness of our experiment by comparing with the other network protocols such as CC-Link [36] which allows to connect remote I/O function modules, intelligent function modules, specific function modules, and so on. It is an industrial open network that enables communication among devices. In those tests, one Programmable Logic Controller (PLC) master communicates with an external microprocessor by CC-Link protocol. The output performance of our tests is evaluated with CC-Link's test in order to insist on the real-time response in the industrial system. Table 4 summarizes the competitive results between two network protocols when the same operation characteristics are maintained. Examining the data in more detail, the total transmission

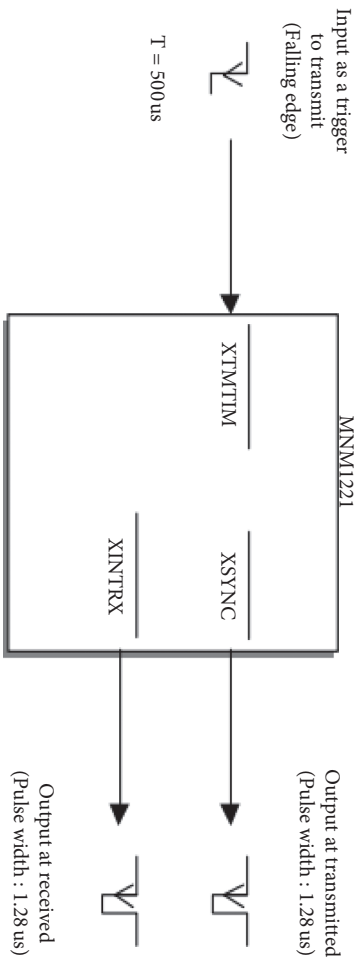


Figure 10: XSYNC signal and other peripheral communication.

Table 4: Competitive performance between our protocol, CC-Link, and CANopen protocol.

Data frame	Our protocol	CC-Link	CANopen
32 bytes	0.0625 ms	0.258 ms	—
4 bytes	—	—	0.4367 ms

and reception are completed in a cycle time. Therefore, for one data package, our protocol takes 0.0625 ms while CC-Link protocol spends 0.258 ms. Since the transmission rate in those CC-Link tests is only 10 Mbps, our protocol provides the superior performance when data are transferred at 100 Mbps. Once, a popular protocol, for example, CANopen [31], is mentioned again to insist on the excellent response of our approach.

Dissimilar to IT (Information Technology) security, system security for industrial network was not given much attention. The reason might be small number of users who regularly come from industry. Consequently, the vulnerability assessments in the industrial network could be reported and analyzed. In addition, the redundancy or latency in this protocol was not deeply evaluated in some research studies. Context-awareness is also a promising feature that developers expect to integrate in further studies. The advanced schemes such artificial intelligence or machine learning algorithm are embedded to investigate network model that comprise data gathering, parsing, and training.

6. Conclusions

In this paper, a modular design of real-time network controller has been developed successfully. The analyses of network protocol characteristic as well as technical specifications were explained in detail. The testification of circuit schematic and electric connections provides selectable operating mode. In firmware level, the flow of data exchange should be forced to update fluently and continuously. The sampling frequency is 25 MHz same as driving the physical IC.

The contributions in this work are (1) a compact design of module for the network motion controller, (2) both hardware schematic and software implementation using the real-time express protocol, and (3) practical validation in the closed-loop servo system. From laboratory experiment, it can be clearly seen that the proposed module can meet the control constraints in motion system, automation factory, and robotics system.

Future work is a must to achieve the significantly high performance; the embedded system should be upgraded, for instance, more advanced microprocessor that could handle many complex computations. The external bus might be optionally Peripheral Component Interconnect (PCI) port or PCI Extended (PCI-e) port for faster communication. Besides, maximum number of slaves ought to be installed in the ring topology in order to gain the optimal results.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

We acknowledge the support of time and facilities from Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for this study.

References

[1] H. Dou, "Lyapunov approach for motion synchronization of a two-slider system," *Transactions of the Institute of Measurement and Control*, vol. 41, no. 14, pp. 4063–4072, 2019.

[2] H. K. Abdul-Ameer, L. I. Abdul-Kreem, H. Adnan, and Z. Sami, "A haptic feedback system based on leap motion controller for prosthetic hand application," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 6, 2020.

[3] T. Li, W. A. Zhang, and L. Yu, "Improved switched system Approach to networked control systems with time-varying delays," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 6, pp. 2711–2717, 2018.

[4] S. Jamian, S. N. S. Salim, M. N. Kamarudin et al., "Review on controller design in pneumatic actuator drive system,"

TELKOMNIKA (Telecommunication Computing Electronics and Control), vol. 18, no. 1, pp. 332–342, 2020.

[5] J. Huang and Q.-G. Wang, "Decentralized adaptive control of interconnected nonlinear systems with unknown control directions," *ISA Transactions*, vol. 74, pp. 60–66, 2018.

[6] W. Liang, Y. Fan, K.-C. Li, D. Zhang, and J.-L. Gaudiot, "Secure data storage and recovery in industrial blockchain network environments," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6543–6552, 2020.

[7] X. Xu, G.-Y. Gu, Z. Xiong, X. Sheng, and X. Zhu, "Development of a decentralized multi-axis synchronous control approach for real-time networks," *ISA Transactions*, vol. 68, pp. 116–126, 2017.

[8] Y. Liu, S. Xie, and Y. Zhang, "Cooperative offloading and resource management for UAV-enabled mobile edge computing in power IoT system," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, Article ID 12229, 2020.

[9] D. Ding, Q.-L. Han, Z. Wang, and X. Ge, "A survey on model-based distributed control and filtering for industrial cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 2483–2499, 2019.

[10] F. Fareeza, C. Rambabu, S. Krishnaveni, and A. C. Kabiso, "Automation of DMPS manufacturing by using LabView & PLC," *International Journal of Electrical and Computer Engineering*, vol. 8, no. 6, p. 5484, 2018.

[11] H. Q.-T. Ngo, T. P. Nguyen, T. S. Le, V. N.-S. Huynh, and H. A.-M. Tran, "Experimental design of PC-based servo system," in *Proceedings of the 2017 International Conference on System Science and Engineering (ICSSE)*, pp. 733–738, Ho Chi Minh City, Vietnam, July 2017.

[12] N. Khalil and A. Najid, "Performance analysis of 802.11ac with frame aggregation using NS3," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 5, p. 5368, 2020.

[13] X. Ge, I. Ahmad, Q.-L. Han, J. Wang, and X.-M. Zhang, "Dynamic event-triggered scheduling and control for vehicle active suspension over controller area network," *Mechanical Systems and Signal Processing*, vol. 152, Article ID 107481, 2021.

[14] R. M. Patil and N. R. Vinay, "Application-based smart parking system using CAN bus," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 12, no. 2, pp. 759–764, 2018.

[15] D. Shah and A. Mehta, "Discrete-time sliding mode controller subject to real-time fractional delays and packet losses for networked control system," *International Journal of Control, Automation and Systems*, vol. 15, no. 6, pp. 2690–2703, 2017.

[16] M. I. Rusydi, S. Syafii, R. Hadelina, E. Kimin, A. W. Setiawan, and A. Rusydi, "Recognition of sign language hand gestures using leap motion sensor based on threshold and ANN models," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 2, pp. 473–483, 2020.

[17] J. Wu, D. Li, and S. Wang, "The design and experimental research of an open architecture soft-CNC system based on RTX and an IPC," *International Journal of Advanced Manufacturing Technology*, vol. 89, no. 5, pp. 1387–1399, 2017.

[18] W. Hang Suan, A. Bahari Jambek, M. N. Bin Md Isa, A. bin Harun, S. N. Bin Mohyar, and Z. A. B. A. Aziz, "Design and implementation of AMBA bridge protocol in system on chip design," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 2, pp. 788–795, 2019.

[19] H. Ngo, Q. Nguyen, and W. Kim, "Implementation of input shaping control to reduce residual vibration in industrial network motion system," in *Proceedings of the 2015 Fiftyth International Conference on Control, Automation and Systems (ICCAS)*, pp. 1693–1698, Busan, Republic of Korea, October 2015.

[20] C. C. Leong and M. K. Ishak, "Performance evaluation of embedded ethernet and Controller Area Network (CAN) in real time control communication system," *International Journal of Reconfigurable and Embedded Systems*, vol. 8, no. 1, p. 36, 2019.

[21] B. K. Seljak, "Hardware-software co-design for a real-time executive," in *Proceedings of the IEEE International Symposium on Industrial Electronics*, pp. 55–58, Bled, Slovenia, July 1999.

[22] M. A. S. Aboelela, "Application of optimal artificial intelligence based tuned controllers to a class of embedded nonlinear power system," *International Journal of Reconfigurable and Embedded Systems*, vol. 9, no. 1, p. 83, 2020.

[23] H. Q. T. Ngo, Q. C. Nguyen, and T. P. Nguyen, "Design and implementation of high-performance motion controller for 2-D delta robot," in *Proceedings of the 2017 Seventh International Conference On Information Science And Technology (ICIST)*, pp. 129–134, Da Nang, Vietnam, April 2017.

[24] M. I. Sani, S. Siregar, M. M. Kurnia, and D. Hasbialloh, "An electrical power control system for explorer-class remotely operated underwater vehicle (ROV)," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 17, no. 2, p. 928, 2019.

[25] R. Li, S. Lin, and J. Chen, "Embedded Motion Controller Design Based on RTEX Network," in *Proceedings of the Fifth International Conference on Intelligent Human-Machine Systems and Cybernetics*, pp. 326–329, Hangzhou, China, August 2013.

[26] C. B. Lim and M. H. Salih, "Design and implementation of embedded concurrent laser missile jammer system using FPGA," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 2, pp. 780–787, 2019.

[27] Y.-M. Yeon and S.-H. Kim, "Design of high-speed easy-to-expand CC-Link parallel communication module based on R-IN32M3," *IEICE - Transactions on Info and Systems*, vol. E102.D, no. 1, pp. 116–123, 2019.

[28] C. W. Cheng and W. P. Tseng, "Design and implementation of a real-time NURBS surface interpolator," *International Journal of Advanced Manufacturing Technology*, vol. 30, no. 1-2, p. 98, 2006.

[29] A. L. Dias, G. S. Sestito, and D. Brandao, "Performance analysis of profibus dp and profinet in a motion control application," *Journal of Control, Automation and Electrical Systems*, vol. 28, no. 1, pp. 86–93, 2017.

[30] A. Khaliq, S. Saha, B. Bhatt, D. Gu, and K. McDonald-Maier, "Profi-load: an FPGA-based solution for generating network load in profinet communication," in *Proceedings of the 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pp. 865–872, IEEE, Bari, Italy, October 2019.

[31] G. Cena and A. Valenzano, "A protocol for automatic node discovery in CANopen networks," *IEEE Transactions on Industrial Electronics*, vol. 50, no. 3, pp. 419–430, 2003.

[32] J. Liu, T. Wang, Y. Ling, and D. Xu, "A bus protocol application in the network based AC servo system," in *Proceedings of the 2007 Second IEEE Conference on Industrial Electronics and Applications*, pp. 2679–2685, IEEE, Harbin, China, May 2007.

[33] X. Chen, D. Li, J. Wan, and N. Zhou, "A clock synchronization method for EtherCAT master," *Microprocessors and Microsystems*, vol. 46, pp. 211–218, 2016.

[34] K. Langlois, T. van der Hoeven, D. Rodriguez Cianca et al., "Ethercat tutorial: an introduction for real-time hardware

communication on windows [tutorial]," *IEEE Robotics and Automation Magazine*, vol. 25, no. 1, pp. 22–122, 2018.

[35] Panasonic Electric Works Europe AG, "MINAS A6N Servoantriebsregler (RTEX-Netzwerktyp)," 2020, https://www.panasonic-electric-works.com/at/minas-a6n-servoantriebsregler.htm.

[36] J. Yu and J. Wu, "Research on network of communication stations in CC-link vibration friction welding system," in *Proceedings of the 2019 Third International Conference on Electronic Information Technology and Computer Engineering (EITCE)*, pp. 101–107, IEEE, Xiamen, China, October 2019.

Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 5072487, 10 pages
https://doi.org/10.1155/2022/5072487

Research Article

Robust International Portfolio Optimization with Worst-Case Mean-LPM

Fei Luan, Weiguo Zhang, and Yongjun Liu

School of Business Administration, South China University of Technology, Guangzhou 510641, China

Correspondence should be addressed to Weiguo Zhang; wgzhang@scut.edu.cn

Received 8 October 2021; Accepted 12 January 2022; Published 23 February 2022

Academic Editor: Guoqiang Wang

This paper proposes a robust international portfolio optimization model with the consideration of worst-case lower partial moment (LPM) and worst-case mean return. In our model, we assume that the distributions and the first- and second-order moments of distributions of returns of assets and exchange rates are all ambiguous. The proposed model can be reformulated into an equivalent semidefinite programming (SDP) problem, which is computationally tractable. For investigation of the performance of our model, we also give two benchmark models. The first benchmark model is a scenario-based model which uses historical observations of returns to approximate the future distributions. The second benchmark model only considers the ambiguity of distributions but does not consider the ambiguity of the first- and second-order moments of distributions. We conduct empirical experiments in a rolling forward way to evaluate the out-of-sample performances of our proposed model, the two benchmark models, and an equally weighted model using the return measures and various risk-adjusted return measures. The result shows that our model has the best performance. It verifies that investors can obtain benefits when employing the robust model and considering the ambiguity of the first- and second-order moments of distributions.

1. Introduction

In order to capture the diversification benefits of international financial markets, institutional and individual investors tend to invest part of their money in the financial markets of other countries or regions using different currencies. The correlations of returns of assets in other countries or regions are often lower than those in just one country, so international asset allocation may reduce risk [1–5]. Generally, the distribution of international portfolio return is asymmetry. It is well known that variance is not an appropriate measure to evaluate the risk of asymmetry distributions, whereas downside risk measures can measure the risk of asymmetry distributions effectively. Among various downside risk measures, lower partial moments (LPM) are comprehensive and sensible [6]. The definition of LPM is introduced by Bawa [7], Bawa and Lindenberg [8], and Fishburn [9]. In order to compute LPM, we need to know the distributions of future security returns beforehand. However, for investors, the distributions of future security returns are usually unknown or cannot be estimated accurately. Even if they acquire the actual distributions of future security returns, the computation of LPM is also a difficult task. To deal with these problems, some researchers employ robust optimization techniques to portfolio selection models using LPM as the risk measure [10–14]. Their researches focus on the portfolio selection problems in one country and do not consider the international portfolio selection problems with the risk of exchange rates. Meanwhile, their researches only consider the worst-case LPM and do not consider the worst-case mean return. Intuitively, the worst-case mean return can also give investors helpful instructions for making investment decisions.

In this paper, we build a robust international portfolio optimization model with worst-case LPM as the risk measure and consider the worst-case mean return. We assume that the distributions and the first- and second-order moments of distributions of future returns of assets and exchange rates are all ambiguous. Using robust optimization techniques, we reformulate our model into an equivalent

semidefinite programming (SDP) problem. In order to evaluate the performance of our model, we also give two benchmark models. In the first benchmark model, we use historical return observations to form empirical distributions of future returns and build an international portfolio optimization model based on these empirical distributions. In the second benchmark model, we assume that the distributions of future returns are ambiguous, but the first- and second-order moments are known. Then, we conduct empirical experiments using the return measures and various risk-adjusted return measures to assess the performances of our model, the two benchmark models, and the equally weighted model.

This paper is organized as follows. In Section 2, we propose the robust international portfolio optimization model with worst-case LPM and mean return under a distributional ambiguity set where the distributions and the first- and second-order moments are ambiguous. We derive an equivalent SDP reformulation of this model. In Section 3, we present the two benchmark models. In Section 4, we conduct empirical experiments to evaluate the performance of our model in comparison with the two benchmark models and the equally weighted model using the return measures and various risk-adjusted return measures. Section 5 gives the conclusions of this paper.

1.1. Notation. In this paper, vectors are denoted by lowercase boldface letters, and matrices are denoted by uppercase boldface letters. We use $\mathbb{R}^n$ to denote the space of vectors of real numbers with dimension n and $\mathbb{S}^n$ to denote the space of symmetric matrices with dimension n. For any two matrices $\mathbf{X}, \mathbf{Y} \in \mathbb{S}^n$, we use $<\mathbf{X}, \mathbf{Y}> = \text{trace}(XY)$ to denote the trace scalar product, and the relation $\mathbf{X} \succcurlyeq \mathbf{Y}$ represents that $\mathbf{X} - \mathbf{Y}$ is positive semidefinite. Random variables are denoted by symbols with tildes, whereas the realizations of them are denoted by symbols without tildes.

2. Model Formulation

In the international financial markets, we assume that an investor plans to invest in the stock markets of n foreign countries or overseas regions where people use different currencies to the investor's domestic currency. We denote that the return of the asset in the i-th country or region is $\widetilde{s}_i$, and the return of the exchange rate of the i-th country or region is $\widetilde{c}_i$, where $i = 1, 2, \ldots, n$. Then, the return of the i-th asset in the investor's domestic currency can be obtained as

$$\left(1 + \widetilde{s}_i\right)\left(1 + \widetilde{c}_i\right) - 1 = \widetilde{s}_i + \widetilde{c}_i + \widetilde{s}_i\widetilde{c}_i. \tag{1}$$

We assume that the weight of money in the domestic currency of the investor invested in the i-th asset is w_i, and the sum of w_i, $i = 1, 2, \ldots, n$, equals 1. Then, the total return of the international portfolio $\mathbf{w} = (w_1, w_2, \ldots, w_n)^T$ can be written as

$$\sum_{i=1}^{n} \left(\widetilde{s}_i + \widetilde{c}_i + \widetilde{s}_i\widetilde{c}_i\right)w_i. \tag{2}$$

For convenience, we denote that

$$\widetilde{\boldsymbol{\xi}} = \left(\widetilde{s}_1, \widetilde{s}_2, \ldots, \widetilde{s}_n, \widetilde{c}_1, \widetilde{c}_2, \ldots, \widetilde{c}_n\right)^T, \tag{3}$$

which combines the returns of assets and exchange rates in one vector. We also denote that

$$\mathbf{W} = \begin{bmatrix} w_1 & & & \\ & w_2 & & \\ & & \ddots & \\ & & & w_n \end{bmatrix},$$

$$\mathbf{H} = \begin{bmatrix} 0_{n\times n} & \frac{1}{2}\mathbf{W} \\ \frac{1}{2}\mathbf{W} & 0_{n\times n} \end{bmatrix}, \tag{4}$$

$$\mathbf{B}_1 = \begin{bmatrix} \mathbf{I}_{n\times n} \\ 0_{n\times n} \end{bmatrix},$$

$$\mathbf{B}_2 = \begin{bmatrix} 0_{n\times n} \\ \mathbf{I}_{n\times n} \end{bmatrix}.$$

Then, (2) can be rewritten as

$$r\left(\mathbf{w}, \widetilde{\boldsymbol{\xi}}\right) = \widetilde{\boldsymbol{\xi}}^T\mathbf{B}_1\mathbf{w} + \widetilde{\boldsymbol{\xi}}^T\mathbf{B}_2\mathbf{w} + \widetilde{\boldsymbol{\xi}}^T\mathbf{H}\widetilde{\boldsymbol{\xi}}. \tag{5}$$

We employ first-order lower partial moment to measure the risk of international portfolios. For a given benchmark return a, the first-order LPM can be written as follows:

$$\text{LPM}\left(\mathbf{w}, P\right) = E_P\left[\left(a - r\left(\mathbf{w}, \widetilde{\boldsymbol{\xi}}\right)\right)_+\right], \tag{6}$$

where P is the distribution of $\widetilde{\xi}$. In practice, investors usually cannot know P accurately beforehand. Thus, some researchers use historical observations of returns to form an empirical approximation of P. We assume that investors can obtain m historical observations, which are denoted by $\widehat{\xi}_1, \widehat{\xi}_2, \ldots, \widehat{\xi}_m$. The empirical approximation $\widehat{P}$ of P is typically formed as follows:

$$\widehat{P}\left(\widetilde{\boldsymbol{\xi}} = \widehat{\boldsymbol{\xi}}_i\right) = \frac{1}{m}, \quad i = 1, 2, \ldots, m. \tag{7}$$

Under $\widehat{P}$, the LPM in (6) can be rewritten as

$$\text{LPM}\left(\mathbf{w}, \widehat{P}\right) = \sum_{i=1}^{m} \frac{1}{m}\left[\left(a - r\left(\mathbf{w}, \widehat{\xi}_i\right)\right)_+\right]. \tag{8}$$

If m is relatively large, according to the law of large numbers, the gap between (6) and its scenario-based version (8) can be small. But the number of observations that investors can acquire is usually small and cannot satisfy the requirement of the law of large numbers. Instead, some researchers use historical observations of returns to form a distributional ambiguity set of future returns. A popular ambiguity set that considers the ambiguity of the

distributions and the first- and second-order moments of distributions is proposed by Delage and Ye [15]. This ambiguity set denoted by $\mathscr{P}_1$ can be described as follows:

$$\mathscr{P}_1\left(\widehat{\boldsymbol{\mu}}, \widehat{\boldsymbol{\Sigma}}, \lambda_1, \lambda_2\right) = \left\{ \begin{array}{c} \mathbb{P}\left(\widetilde{\boldsymbol{\xi}} \in \mathbb{R}^{2n}\right) = 1 \\ P \in \mathscr{M} \mid \left(\mathbb{E}\left(\widetilde{\boldsymbol{\xi}}\right) - \widehat{\mu}\right)^T \widehat{\boldsymbol{\Sigma}}^{-1} \left(\mathbb{E}\left(\widetilde{\boldsymbol{\xi}}\right) - \widehat{\boldsymbol{\mu}}\right) \leq \lambda_1 \\ \mathbb{E}\left[\left(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}\right)\left(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}\right)^T\right] \leqq \lambda_2 \widehat{\boldsymbol{\Sigma}} \end{array} \right\}, \tag{9}$$

where $\mathscr{M}$ is the set of all probability measures on the measurable space $(\mathbb{R}^{2n}, \mathscr{B})$, with $\mathscr{B}$ being the Borel σ-algebra on $\mathbb{R}^{2n}$, $\widehat{\mu}$ is the sample-based mean return, $\widehat{\Sigma}$ is the sample-based covariance matrix, λ_1 reflects the ambiguity size of mean return, and λ_2 reflects the ambiguity size of the covariance matrix. The worst-case LPM with respect to the ambiguity set $\mathscr{P}_1$ denoted by $WLPM(\mathbf{w}, \mathscr{P}_1)$ can be defined as

$$\mathrm{WLPM}\left(\mathbf{w}, \mathscr{P}_1\right) = \max_{P \in \mathscr{P}_1} \mathrm{LPM}\left(\mathbf{w}, \mathscr{P}_1\right). \tag{10}$$

Since the mean return of the international portfolio can also give investors useful instructions for decision-making, we add the worst-case mean return in the objective function of our model. We give the definition of worst-case mean return with respect to $\mathscr{P}_1$ denoted by $W\mathrm{Return}(\mathbf{w}, \mathscr{P}_1)$ as follows:

$$W\mathrm{Return}\left(\mathbf{w}, \mathscr{P}_1\right) = \min_{P \in \mathscr{P}_1} E_P\left[r\left(\mathbf{w}, \widetilde{\boldsymbol{\xi}}\right)\right]. \tag{11}$$

The robust international portfolio optimization model using worst-case LPM as the risk measure and considering worst-case mean return under $\mathscr{P}_1$ can be built as follows:

$$(\mathrm{RIML}) \min_{\mathbf{w}} \lambda \cdot \mathrm{WLPM}\left(\mathbf{w}, \mathscr{P}_1\right) - (1 - \lambda) \cdot W\,\mathrm{Return}\left(\mathbf{w}, \mathscr{P}_1\right). \tag{12a}$$

$$\text{s.t. } \mathbf{w} \in \mathbb{R}^n, \mathbf{w}^T \mathbf{e} = 1, \mathbf{w} \geq 0, \tag{12b}$$

where λ is the risk aversion coefficient of investors and $\mathbf{e}$ denotes the vector of 1 s with dimension n. Problem (12) cannot be solved directly; thus, we need to derive its equivalent reformulation, which is computationally tractable. In the following, we first give the equivalent SDP reformulation of $WLPM(\mathbf{w}, \mathscr{P}_1)$ defined by (10); then, we give the equivalent SDP reformulation of $WReturn(\mathbf{w}, \mathscr{P}_1)$ defined by (11).

Theorem 1. *$WLPM(\mathbf{w}, \mathscr{P}_1)$ defined by (10) is equal to the optimal objective function value of the following SDP problem:*

$$\min_{\{p_1, \mathbf{U}, \mathbf{V}, \mathbf{v}, s\}} p_1 + \lambda_2 \langle \mathbf{U}, \widehat{\Sigma} \rangle, \tag{13a}$$

$$\text{s.t.} \mathbf{U} \succcurlyeq 0, \tag{13b}$$

$$\begin{bmatrix} \mathbf{V} & \mathbf{v} \\ \mathbf{v}^T & s \end{bmatrix} \succcurlyeq 0, \tag{13c}$$

$$\begin{bmatrix} \mathbf{U} & \frac{1}{2}\left(-2\mathbf{U}\widehat{\boldsymbol{\mu}} - 2\mathbf{v}\right) \\ \frac{1}{2}\left(-2\mathbf{U}\widehat{\boldsymbol{\mu}} - 2\mathbf{v}\right)^T & p_1 + \widehat{\boldsymbol{\mu}}^T \mathbf{U} \widehat{\boldsymbol{\mu}} - \langle \mathbf{V}, \widehat{\boldsymbol{\Sigma}} \rangle + 2\mathbf{v}^T \widehat{\boldsymbol{\mu}} - \lambda_1 s \end{bmatrix} \succcurlyeq 0, \tag{13d}$$

$$\begin{bmatrix} \mathbf{U} + \mathbf{H} & \frac{1}{2}\left(-2\mathbf{U}\widehat{\boldsymbol{\mu}} - 2\mathbf{v} + \mathbf{B}_1\mathbf{w} + \mathbf{B}_2\mathbf{w}\right) \\ \frac{1}{2}\left(-2\mathbf{U}\widehat{\boldsymbol{\mu}} - 2\mathbf{v} + \mathbf{B}_1\mathbf{w} + \mathbf{B}_2\mathbf{w}\right)^T & p_1 + \widehat{\boldsymbol{\mu}}^T \mathbf{U} \widehat{\boldsymbol{\mu}} - \langle \mathbf{V}, \widehat{\boldsymbol{\Sigma}} \rangle + 2\mathbf{v}^T \widehat{\boldsymbol{\mu}} - \lambda_1 s - a \end{bmatrix} \succcurlyeq 0. \tag{13e}$$

Proof. $\mathrm{WLPM}(\mathbf{w}, \mathscr{P}_1)$ defined by (10) is equivalent to the following problem:

$$-\min_{P \in \mathscr{P}_1} -E_P\left[\left(a - r\left(\mathbf{w}, \widetilde{\boldsymbol{\xi}}\right)\right)_+\right]. \tag{14}$$

Then, we shall derive the equivalent SDP reformulation of the following:

$$\min_{P \in \mathscr{P}_1} -E_P\left[\left(a - r\left(\mathbf{w}, \widetilde{\boldsymbol{\xi}}\right)\right)_+\right], \tag{15}$$

which can be written as

$$\min_P -\int_{\mathbb{R}^{2n}} \left(a - r\left(\mathbf{w}, \widetilde{\boldsymbol{\xi}}\right)\right)_+ dP\left(\widetilde{\boldsymbol{\xi}}\right), \tag{16a}$$

$$\text{s.t.} \int_{\mathbb{R}^{2n}} \mathrm{dP}\left(\widetilde{\boldsymbol{\xi}}\right) = 1, \tag{16b}$$

$$\int_{\mathbb{R}^{2n}} \left(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}\right)\left(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}\right)^T dP\left(\widetilde{\boldsymbol{\xi}}\right) \prec \lambda_2 \widehat{\boldsymbol{\Sigma}}, \tag{16c}$$

$$\int_{\mathbb{R}^{2n}} \begin{bmatrix} \widehat{\Sigma} & \left(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}\right) \\ \left(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}\right)^T & \lambda_1 \end{bmatrix} dP\left(\widetilde{\boldsymbol{\xi}}\right) \succcurlyeq 0, \tag{16d}$$

$$P \in \mathscr{M}. \tag{16e}$$

We denote the dual variable of constraint (16b) by p_1, that of constraint (16c) by $\mathbf{U}$, and that of constraint (16d) by

$$\begin{bmatrix} \mathbf{V} & v \\ \mathbf{v}^T & s \end{bmatrix}, \tag{17}$$

where $p_1 \in \mathbb{R}$; $\mathbf{U} \in \mathbb{S}^{2n}$; $\mathbf{U} \succcurlyeq 0$; $\mathbf{V} \in \mathbb{S}^{2n}$; $\mathbf{v} \in \mathbb{R}^{2n}$; $s \in \mathbb{R}$;

$$\begin{bmatrix} \mathbf{V} & v \\ \mathbf{v}^T & s \end{bmatrix} \succcurlyeq 0. \tag{18}$$

The dual reformulation of problem (16a) can be written as follows:

$$\max_{\{p_1,\mathbf{U},\mathbf{V},\mathbf{v},s\}} -p_1 - \lambda_2 \langle \mathbf{U}, \widehat{\Sigma} \rangle, \tag{19a}$$

$$\begin{aligned} \text{s.t. } & p_1 + (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^T \mathbf{U} (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) - \langle \mathbf{V}, \widehat{\Sigma} \rangle - 2\mathbf{v}^T (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) \\ & - \lambda_1 s \geq (a - r(\mathbf{w}, \widetilde{\boldsymbol{\xi}}))_+, \quad \forall \widetilde{\boldsymbol{\xi}} \in \mathbb{R}^{2n}. \end{aligned} \tag{19b}$$

The Dirac measure $\delta_{\widehat{\mu}}$ is the measure of mass one at the point $\widehat{\mu}$. Obviously, $\delta_{\widehat{\mu}}$ lies in the relative interior of the feasible set of problem (16). According to the weak version of Proposition 3.4 in Shapiro [16], we can deduce that there is no dual gap between problems (16) and (19). Thus, the optimal objective function value of problem (16) is equal to that of problem (19). Constraint (19b) is equivalent to the following two constraints:

$$\begin{aligned} & p_1 + (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^T \mathbf{U} (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) - \langle \mathbf{V}, \widehat{\Sigma} \rangle - 2\mathbf{v}^T (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) \\ & \quad - \lambda_1 s \geq 0, \quad \forall \widetilde{\boldsymbol{\xi}} \in \mathbb{R}^{2n}. \end{aligned} \tag{20}$$

$$\begin{aligned} & p_1 + (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^T \mathbf{U} (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) - \langle \mathbf{V}, \widehat{\Sigma} \rangle - 2\mathbf{v}^T (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) \\ & \quad - \lambda_1 s \geq a - r(\mathbf{w}, \widetilde{\boldsymbol{\xi}}), \quad \forall \widetilde{\boldsymbol{\xi}} \in \mathbb{R}^{2n}. \end{aligned} \tag{21}$$

Equation (20) is equivalent to the matrix inequality (13d), and (21) is equivalent to the matrix inequality (13e). Thus, we complete the proof of this theorem. □

Theorem 2. *$W\mathrm{Return}(\mathbf{w}, \mathscr{P}_1)$ defined in (11) is equal to the optimal objective function value of the following SDP problem:*

$$\max_{\{q_1,\mathbf{Q},\mathbf{M},\mathbf{h},z\}} -q_1 - \lambda_2 \langle \mathbf{Q}, \widehat{\Sigma} \rangle, \tag{22a}$$

$$\text{s.t.} \mathbf{Q} \succcurlyeq 0, \tag{22b}$$

$$\begin{bmatrix} \mathbf{M} & \mathbf{h} \\ \mathbf{h}^T & z \end{bmatrix} \succcurlyeq 0, \tag{22c}$$

$$\begin{bmatrix} \mathbf{H} + \mathbf{Q} & \frac{1}{2}(\mathbf{B}_1\mathbf{w} + \mathbf{B}_2\mathbf{w} - 2\mathbf{Q}\widehat{\boldsymbol{\mu}} - 2\mathbf{h}) \\ \frac{1}{2}(\mathbf{B}_1\mathbf{w} + \mathbf{B}_2\mathbf{w} - 2\mathbf{Q}\widehat{\boldsymbol{\mu}} - 2\mathbf{h})^T & q_1 + \widehat{\boldsymbol{\mu}}^T\mathbf{Q}\widehat{\boldsymbol{\mu}} - \langle \mathbf{M}, \widehat{\boldsymbol{\Sigma}} \rangle + 2\mathbf{h}^T\widehat{\boldsymbol{\mu}} - \lambda_1 z \end{bmatrix} \succcurlyeq 0. \tag{22d}$$

Proof: . $W\mathrm{Return}(\mathbf{w}, \mathscr{P}_1)$ defined by (11) can be written in the following formulation:

$$\min_P \int_{\mathbb{R}^{2n}} \left(\widetilde{\boldsymbol{\xi}}^T \mathbf{B}_1 \mathbf{w} + \widetilde{\boldsymbol{\xi}}^T \mathbf{B}_2 \mathbf{w} + \widetilde{\boldsymbol{\xi}}^T \mathbf{H} \widetilde{\boldsymbol{\xi}} \right) dP(\widetilde{\xi}), \tag{23a}$$

$$\text{s.t.} \int_{\mathbb{R}^{2n}} dP(\widetilde{\boldsymbol{\xi}}) = 1, \tag{23b}$$

$$\int_{\mathbb{R}^{2n}} (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})(\widetilde{\boldsymbol{\xi}} - \widehat{\mu})^T dP(\widetilde{\boldsymbol{\xi}}) \prec \lambda_2 \widehat{\boldsymbol{\Sigma}}, \tag{23c}$$

$$\int_{\mathbb{R}^{2n}} \begin{bmatrix} \widehat{\boldsymbol{\Sigma}} & (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) \\ (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^T & \lambda_1 \end{bmatrix} dP(\widetilde{\boldsymbol{\xi}}) \succcurlyeq 0, \tag{23d}$$

$$P \in \mathscr{M}. \tag{23e}$$

Similar to the proof of Theorem 1, we denote the dual variable of constraint (23b) by q_1, that of constraint (16c) by Q, and that of constraint (16d) by

$$\begin{bmatrix} \mathbf{M} & \mathbf{h} \\ \mathbf{h}^T & z \end{bmatrix}, \tag{24}$$

where $q_1 \in \mathbb{R}$; $\mathbf{Q} \in \mathbb{S}^{2n}$; $\mathbf{Q} \succcurlyeq 0$; $\mathbf{M} \in \mathbb{S}^{2n}$; $\mathbf{h} \in \mathbb{R}^{2n}$; $z \in \mathbb{R}$;

$$\begin{bmatrix} \mathbf{M} & \mathbf{h} \\ \mathbf{h}^T & z \end{bmatrix} \succcurlyeq 0. \tag{25}$$

The dual reformulation of problem (16) can be written as

$$\max_{\{q_1,\mathbf{Q},\mathbf{M},\mathbf{h},z\}} -q_1 - \lambda_2 \langle \mathbf{Q}, \widehat{\Sigma} \rangle, \tag{26a}$$

$$\text{s.t.} \quad \mathbf{Q} \succcurlyeq 0, \tag{26b}$$

$$\begin{bmatrix} \mathbf{M} & \mathbf{h} \\ \mathbf{h}^T & z \end{bmatrix} \succcurlyeq 0, \tag{26c}$$

$$\begin{aligned} & \widetilde{\boldsymbol{\xi}}^T \mathbf{B}_1 \mathbf{w} + \widetilde{\boldsymbol{\xi}}^T \mathbf{B}_2 \mathbf{w} + \widetilde{\boldsymbol{\xi}}^T \mathbf{H} \widetilde{\boldsymbol{\xi}} + q_1 + (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^T \mathbf{Q} (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) \\ & \quad - \langle \mathbf{M}, \widehat{\boldsymbol{\Sigma}} \rangle - 2\mathbf{h}^T (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) - \lambda_1 z \geq 0, \quad \forall \widetilde{\boldsymbol{\xi}} \in \mathbb{R}^{2n}. \end{aligned} \tag{26d}$$

Similar to the proof of Theorem 1, there is no dual gap between problems (23) and (26); thus, the optimal objective function values of the two problems are the same. The equivalent matrix inequality of constraint (26d) is (22d); thus, we complete the proof of this theorem.

With Theorems 1 and 2, we can easily obtain the equivalent SDP reformulation of problem RIML (12), and the final formulation is as follows:

$$\min_{\left\{\begin{array}{l} p_1, \mathbf{U}, \mathbf{V}, \mathbf{v}, s, \\ q_1, \mathbf{Q}, \mathbf{M}, \mathbf{h}, z \end{array}\right\}} \lambda \cdot \left(p_1 + \lambda_2 \langle \mathbf{U}, \widehat{\Sigma} \rangle\right) + (1-\lambda) \cdot \left(q_1 + \lambda_2 \langle \mathbf{Q}, \widehat{\Sigma} \rangle\right), \tag{27a}$$

$$\text{s.t. (12b), (13b), (13c), (13d), (13e), (22b), (22c), (22d).} \tag{27b}$$

□

3. Two Benchmark Models

In order to assess the performance of our model RIML, we present two benchmark models in this section. The first benchmark model denoted by SIML is based on empirical distributions approximated by historical samples of returns. The approximated distribution $\widehat{P}$ is described in (7), and LPM under $\widehat{P}$ is shown in (8). The return of the international portfolio under $\widehat{P}$ can be written as

$$\text{Return}(\mathbf{w}, \widehat{P}) = \frac{1}{m} \sum_{i=1}^{m} r\left(\mathbf{w}, \widehat{\boldsymbol{\xi}}_i\right). \tag{28}$$

The scenario-based international portfolio optimization model with mean-LPM denoted by SIML is built as

$$(\text{SIML}) \min_{\mathbf{w}} \lambda \cdot \text{LPM}(\mathbf{w}, \widehat{\text{P}}) - (1-\lambda) \cdot \text{Return}(\mathbf{w}, \widehat{\text{P}}), \tag{29a}$$

$$\text{s.t. } \mathbf{w} \in \mathbb{R}^{\text{n}}, \mathbf{e}^{\text{T}}\mathbf{w} = 1, \mathbf{w} \geq 0, \tag{29b}$$

where $LPM(\mathbf{w}, \widehat{P})$ is defined by (8) and $\text{Return}(\mathbf{w}, \widehat{P})$ is defined by (28). In the second benchmark model denoted by RIML^{-}, we assume that the distributions of returns of assets and exchange rates are ambiguous, but the first- and second-order moments of distributions are determined beforehand. The corresponding ambiguity set $\mathscr{P}_2$ is described as follows:

$$\mathscr{P}_2(\widehat{\boldsymbol{\mu}}, \widehat{\boldsymbol{\Sigma}}) = \left\{ P \in \mathscr{M} \,\middle|\, \begin{array}{c} P\left(\widetilde{\boldsymbol{\xi}} \in \mathbb{R}^{2n}\right) = 1 \\ \mathbb{E}(\widetilde{\boldsymbol{\xi}}) = \widehat{\mu} \\ \mathbb{E}\left[(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^T\right] = \widehat{\boldsymbol{\Sigma}} \end{array} \right\}, \tag{30}$$

where the definitions of $\mathscr{M}$, $\widehat{\boldsymbol{\mu}}$, and $\widehat{\boldsymbol{\Sigma}}$ are the same as those of (9). Under $\mathscr{P}_2$, RIML^{-} can be written as

$$(\text{RIML}^{-}) \min_{\mathbf{w}} \lambda \cdot WLPM(\mathbf{w}, \mathscr{P}_2) - (1-\lambda) \cdot W\text{Return}(\mathbf{w}, \mathscr{P}_2), \tag{31a}$$

$$\text{s.t.} \mathbf{w} \in \mathbb{R}^{n}, \mathbf{w}^{T}\mathbf{e} = 1, \mathbf{w} \geq 0, \tag{31b}$$

where

$$\text{WLPM}(\mathbf{w}, \mathscr{P}_2) = \max_{P \in \mathscr{P}_2} LPM(\mathbf{w}, P), \tag{32}$$

$$W\text{Return}(\mathbf{w}, \mathscr{P}_2) = \min_{P \in \mathscr{P}_2} E_P(r(\mathbf{w}, \widetilde{\boldsymbol{\xi}})). \tag{33}$$

In the following, we also try to derive the equivalent SDP reformulations of $WLPM(\mathbf{w}, \mathscr{P}_2)$ defined by (32) and $W\text{Return}(\mathbf{w}, \mathscr{P}_2)$ defined by (33).

Theorem 3. *$WLPM(\mathbf{w}, \mathscr{P}_2)$ defined by (32) is equal to the optimal objective function value of the following SDP problem:*

$$\min_{\{p_1, \mathbf{v}, \mathbf{U}\}} p_1 + \mathbf{v}^T\widehat{\mu} + \langle \mathbf{U}, \widehat{\Sigma} \rangle, \tag{34a}$$

$$\text{s.t.} \begin{bmatrix} \mathbf{U} + \mathbf{H} & \frac{1}{2}(\mathbf{v} - 2\mathbf{U}\widehat{\boldsymbol{\mu}} + \mathbf{B}_1\mathbf{w} + \mathbf{B}_2\mathbf{w}) \\ \frac{1}{2}(\mathbf{v} - 2\mathbf{U}\widehat{\boldsymbol{\mu}} + \mathbf{B}_1\mathbf{w} + \mathbf{B}_2\mathbf{w})^{\text{T}} & p_1 + \widehat{\boldsymbol{\mu}}^{\text{T}}\mathbf{U}\widehat{\boldsymbol{\mu}} - a \end{bmatrix} \succcurlyeq 0, \tag{34b}$$

$$\begin{bmatrix} \mathbf{U} & \frac{1}{2}(\mathbf{v} - 2\mathbf{U}\widehat{\boldsymbol{\mu}}) \\ \frac{1}{2}(\mathbf{v} - 2\mathbf{U}\widehat{\boldsymbol{\mu}})^T & p_1 + \widehat{\boldsymbol{\mu}}^T\mathbf{U}\widehat{\boldsymbol{\mu}} \end{bmatrix} \succcurlyeq 0. \tag{34c}$$

Proof. According to (32), $WLPM(\mathbf{w}, \mathscr{P}_2)$ can be rewritten as

$$-\min_{P \in \mathscr{P}_2} -E_P\left[(a - r(\mathbf{w}, \widetilde{\boldsymbol{\xi}}))_+\right]. \tag{35}$$

Thus, we first study the equivalent SDP reformulation of the following problem:

$$\min_{P \in \mathscr{P}_2} -E_P\left[(a - r(\mathbf{w}, \widetilde{\boldsymbol{\xi}}))_+\right], \tag{36}$$

which can be rewritten as

$$\min_P -\int_{\mathbb{R}^{2n}} \left[(a - r(\mathbf{w}, \widetilde{\boldsymbol{\xi}}))_+\right] dP(\widetilde{\boldsymbol{\xi}}), \tag{37a}$$

$$\text{s.t.} \int_{\mathbb{R}^{2n}} dP(\widetilde{\boldsymbol{\xi}}) = 1, \tag{37b}$$

$$\int_{\mathbb{R}^{2n}} \widetilde{\boldsymbol{\xi}} dP(\widetilde{\boldsymbol{\xi}}) = \widehat{\boldsymbol{\mu}}, \tag{37c}$$

$$\int_{\mathbb{R}^{2n}} (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^T dP(\widetilde{\boldsymbol{\xi}}) = \widehat{\Sigma}. \tag{37d}$$

We set the dual variable of constraint (37b) as p_1, that of constraint (37c) as $\mathbf{v}$, and that of constraint (37d) as $\mathbf{U}$. The dual reformulation of problem (37) can be written as follows:

$$\max_{\{p_1, \mathbf{v}, \mathbf{U}\}} -p_1 - \mathbf{v}^T\widehat{\mu} - \langle \mathbf{U}, \widehat{\Sigma} \rangle, \tag{38a}$$

$$\text{s.t. } p_1 + \mathbf{v}^{\text{T}}\widetilde{\boldsymbol{\xi}} + (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^{\text{T}}\mathbf{U}(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) \geq (a - r(\mathbf{w}, \widetilde{\boldsymbol{\xi}}))_+, \quad \forall \widetilde{\boldsymbol{\xi}} \in \mathbb{R}^{2n}. \tag{38b}$$

Obviously, there is no dual gap between problems (37) and (38). Thus, the two problems (37) and (38) have the same optimal objective function value. We note that constraint (38b) is equivalent to the following two inequalities:

$$p_1 + \mathbf{v}^T\widetilde{\boldsymbol{\xi}} + (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^T\mathbf{U}(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) \geq a - r(\mathbf{w}, \widetilde{\boldsymbol{\xi}}), \quad \forall\widetilde{\boldsymbol{\xi}} \in \mathbb{R}^{2n}, \tag{39}$$

$$p_1 + \mathbf{v}^T\widetilde{\boldsymbol{\xi}} + (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^T\mathbf{U}(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) \geq 0, \quad \forall\widetilde{\boldsymbol{\xi}} \in \mathbb{R}^{2n}. \tag{40}$$

The equivalent matrix inequality of (39) is (34b), and that of (40) is (34c). Thus, we complete the proof of this theorem. □

Theorem 4. *$W\mathrm{Return}(\mathbf{w}, \mathscr{P}_2)$ defined by (33) is equal to the optimal objective function value of the following SDP problem:*

$$\max_{\{q_1,\mathbf{h},\mathbf{Q}\}} -q_1 - \mathbf{h}^T\widehat{\mu} - \langle\mathbf{Q}, \widehat{\Sigma}\rangle, \tag{41a}$$

$$\text{s.t.} \quad \begin{bmatrix} \mathbf{H} + \mathbf{Q} & \frac{1}{2}(\mathbf{B}_1\mathbf{w} + \mathbf{B}_2\mathbf{w} + \mathbf{h} - 2\mathbf{Q}\widehat{\boldsymbol{\mu}}) \\ \frac{1}{2}(\mathbf{B}_1\mathbf{w} + \mathbf{B}_2\mathbf{w} + \mathbf{h} - 2\mathbf{Q}\widehat{\boldsymbol{\mu}})^{\mathrm{T}} & q_1 + \widehat{\boldsymbol{\mu}}^T\mathbf{Q}\widehat{\boldsymbol{\mu}} \end{bmatrix} \succcurlyeq 0. \tag{41b}$$

Proof. $W\mathrm{Return}(\mathbf{w}, \mathscr{P}_2)$ defined by (33) can be rewritten as follows:

$$\min_P \int_{\mathbb{R}^{2n}} \left(\widetilde{\boldsymbol{\xi}}^T\mathbf{B}_1\mathbf{w} + \widetilde{\boldsymbol{\xi}}^T\mathbf{B}_2\mathbf{w} + \widetilde{\boldsymbol{\xi}}^T\mathbf{H}\widetilde{\boldsymbol{\xi}}\right) dP(\widetilde{\boldsymbol{\xi}}), \tag{42a}$$

$$\text{s.t.} \int_{\mathbb{R}^{2n}} dP(\widetilde{\boldsymbol{\xi}}) = 1, \tag{42b}$$

$$\int_{\mathbb{R}^{2n}} \widetilde{\boldsymbol{\xi}} dP(\widetilde{\boldsymbol{\xi}}) = \widehat{\boldsymbol{\mu}}, \tag{42c}$$

$$\int_{\mathbb{R}^{2n}} (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^T dP(\widetilde{\boldsymbol{\xi}}) = \widehat{\Sigma}. \tag{42d}$$

We denote the dual variable of constraint (42b) by q_1, that of constraint (42c) by $\mathbf{h}$, and that of constraint (42d) by $\mathbf{Q}$. The dual reformulation of problem (42) can be written as

$$\max_{\{q_1,\mathbf{h},\mathbf{Q}\}} -q_1 - \mathbf{h}^T\widehat{\mu} - \langle\mathbf{Q}, \widehat{\Sigma}\rangle, \tag{43a}$$

$$\begin{aligned} \text{s.t.}\ & \widetilde{\boldsymbol{\xi}}^{\mathrm{T}}\mathbf{B}_1\mathbf{w} + \widetilde{\boldsymbol{\xi}}^{\mathrm{T}}\mathbf{B}_2\mathbf{w} + \widetilde{\boldsymbol{\xi}}^{\mathrm{T}}\mathbf{H}\widetilde{\boldsymbol{\xi}} + q_1 + \mathbf{h}^{\mathrm{T}}\widetilde{\boldsymbol{\xi}} \\ & + (\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}})^{\mathrm{T}}\mathbf{Q}(\widetilde{\boldsymbol{\xi}} - \widehat{\boldsymbol{\mu}}) \geq 0, \quad \forall\widetilde{\boldsymbol{\xi}} \in \mathbb{R}^{2n}. \end{aligned} \tag{43b}$$

Similarly, there is no dual gap between problems (42) and (43). Thus, the two problems have the same optimal objective function value. The equivalent matrix inequality of (43b) is (41b). Hence, we finish the proof of this theorem.

According to Theorems 3 and 4, we can obtain the equivalent SDP reformulation of model $RIML^-$ defined by (31), and the final formulation is as follows:

$$\min_{\{p_1,\mathbf{v},\mathbf{U},q_1,\mathbf{h},\mathbf{Q}\}} \lambda \cdot \left(p_1 + \mathbf{v}^T\widehat{\mu} + \langle\mathbf{U}, \widehat{\Sigma}\rangle\right) + (1 - \lambda) \cdot \left(q_1 + \mathbf{h}^T\widehat{\mu} + \langle\mathbf{Q}, \widehat{\Sigma}\rangle\right), \tag{44a}$$

$$\text{s.t. (31b), (34b), (34c), (41b).} \tag{44b}$$

□

4. Empirical Experiments

To investigate the performance of our model RIML, we conduct empirical experiments with comparison to models RIML^-, SIML, and EW, where EW is the equally weighted model. We assume that an investor from the Chinese Mainland wants to invest RMB in four main international stock indexes, which are Nikkei 225 (Japan), Hang Seng Index (Hong Kong), S&P 500 (USA), and FTSE 100 (UK). Market prices and spot exchange rates are from the database Wind (https://www.wind.com.cn/). We use weekly historical returns from March 26, 2004, to July 23, 2021 (883 observations). The in-sample period is set from March 26, 2004, to March 30, 2007, containing 150 historical return observations. The out-of-sample period is set from April 6, 2007, to July 23, 2021, containing 733 historical return observations. All experiments are conducted using MATLAB R2018a on an Intel Core i7 CPU 3.20 GHz desktop with 32 GB of RAM. We use MATLAB interface YALMIP by Lofberg [17]. SDP problems (RIML and RIML^-) are solved by the Mosek package (https://www.mosek.com/), and linear programming problem (SIML) is solved by the Gurobi package (https://www.gurobi.com/).

Based on Delage and Ye [15], to compute the two parameters λ_1 and λ_2 of the ambiguity set of our model RIML in (9), we need to build uncertainty sets of the returns of assets and exchange rates. We first illustrate the sample-based mean returns, standard deviations, and covariance matrix of returns of assets and exchange rates during the in-sample period. Table 1 shows sample-based mean returns and standard deviations. From Table 1, we find that the standard deviations of returns of assets are much larger than those of exchange rates. Table 2 shows a sample-based covariance matrix. From Table 2, we find that the covariance of returns of assets is also much larger than those of exchange rates. Thus, we can conclude that the stock market is more volatile than the currency market. According to this observation, we assume that the size of the uncertainty set of returns of assets is larger than that of exchange rates. Specifically, we set the upper bound of returns of assets as 0.06 and the lower bound of those as −0.06, whereas we set the upper bound of returns of exchange rates as 0.02 and the lower bound of those as −0.02.

We conduct rolling forward experiments to assess the out-of-sample performances of models RIML, RIML^-, SIML, and EW. We want to set the benchmark return a as the realized mean return of model RIML in the in-sample period dynamically, and then $LPM(\mathbf{w}, P)$ in (6) is similar to semivariance. However, if we do not know the benchmark return a beforehand, we cannot compute the portfolio of model RIML; thus, we cannot acquire the realized mean return of model RIML in the in-sample period. Hence, instead, we use the realized mean return of model EW in the in-sample period to approximate that of model RIML in the in-sample period dynamically. Specifically, when the realized mean return of model EW is positive or zero, we set the benchmark return a as three times of it. When the realized mean return of model EW is negative, we set a as a third of it. Now we describe the procedure of our rolling forward

Table 1: Sample-based mean returns and standard deviations of assets and exchange rates during the in-sample period.

Asset	Mean	Standard deviation	Currency	Mean	Standard deviation
N225	0.002887	0.022174	JPYCNY	−0.001154	0.011226
HSI	0.003423	0.020297	HKDCNY	−0.000474	0.001555
SPX	0.001875	0.014803	USDCNY	−0.000458	0.001780
FISE	0.002670	0.014631	GBPCNY	0.000180	0.011438

Table 2: Sample-based covariance matrix of returns of assets and exchange rates during the in-sample period.

Covariance	N225	HSI	SPX	FISE	JPYCNY	HKDCNY	USDCNY	GBPCNY
N225	$4.9167e-4$							
HSI	$2.1037e-4$	$4.1199e-4$						
SPX	$1.6326e-4$	$1.3223e-4$	$2.1912e-4$					
FISE	$1.8179e-4$	$1.2539e-4$	$1.5819e-4$	$2.1408e-4$				
JPYCNY	$-1.8237e-5$	$5.1719e-5$	$1.5598e-6$	$-3.4262e-5$	$1.2603e-4$			
HKDCNY	$2.6835e-7$	$-3.6671e-6$	$6.0277e-8$	$2.2379e-7$	$2.1178e-6$	$2.4194e-6$		
USDCNY	$-6.6258e-8$	$-5.0596e-6$	$-1.4664e-6$	$1.5952e-7$	$3.3600e-7$	$2.4910e-6$	$3.1682e-6$	
GBPCNY	$-4.2339e-6$	$7.0252e-5$	$1.7282e-5$	$-3.3184e-5$	$6.9690e-5$	$1.5398e-6$	$8.5962e-7$	$1.3084e-4$

experiment as follows. First, we use the 150 return observations in the in-sample period from March 26, 2004, to March 30, 2007, to compute the relevant parameters of models RIML and RIML$^-$, and determine the three optimal portfolios of models RIML, RIML$^-$, and SIML. According to the realized returns of assets and exchange rates in the first week during the out-of-sample period, we can compute the realized returns of the above three optimal portfolios and the equally weighted strategy. Then, we move the in-sample period one week forward by adding the new week and delete the first week. Based on the return observations in the new in-sample period, we can also compute the three new optimal portfolios of models RIML, RIML$^-$, and SIML. Using the returns observations in the second week of the out-of-sample period and the optimal portfolios obtained from the in-sample period, we can also derive the realized returns of the four models. We continue this procedure until July 23, 2021. Consequently, we obtain four return series with 733 realized returns of the four models. Based on these four return series, we assess the realized performances of the four models using various performance measures, which are mean return, Sharpe ratio [18], downside Sharpe ratio [19], upside potential and downside risk (UP) ratio [20], Mean/VaR, Mean/CVaR, and the cumulative return at July 23, 2021, where VaR and CVaR are at the 0.95 level, Mean/VaR denotes Mean divided by VaR, and Mean/CVaR denotes Mean divided by CVaR. Mean return and the cumulative return on July 23, 2021, are return measures. Sharpe ratio, downside Sharpe ratio, UP ratio, Mean/VaR, and Mean/CVaR are risk-adjusted return measures. According to Sortino and Meer [20], the UP ratio can be defined as follows:

$$UP = \frac{1/K \sum_{t=1}^{K} \max[0, r_t - \rho_t]}{\sqrt{1/K \sum_{t=1}^{K} (\max[0, \rho_t - r_t])^2}}, \tag{45}$$

where r_t is the realized return of a portfolio at the t-th period and ρ_t is a benchmark return at the same period, $t = 1, 2, \ldots, K$. Obviously, the UP ratio is an appropriate measure to assess the performance of portfolios with asymmetry distributions. Without loss of generality, in our numerical experiments, we set that $\rho_t = 0$, $t = 1, 2, \ldots, K$.

For a portfolio $\mathbf{w}$, $WLPM(\mathbf{w}, \mathscr{P}_1)$ in (10) is much larger than $W\text{Return}(\mathbf{w}, \mathscr{P}_1)$ in (11). Hence, in order to balance $\text{WLPM}(\mathbf{w}, \mathscr{P}_1)$ and $W\text{Return}(\mathbf{w}, \mathscr{P}_1)$ and acquire a portfolio that has a good performance in terms of risk-adjusted return, we should set the risk aversion coefficient λ small. In our empirical experiments, we consider various cases of λ, which are $\lambda = 0.03$, $\lambda = 0.02$, $\lambda = 0.01$, $\lambda = 0.009$, $\lambda = 0.008$, $\lambda = 0.007$, $\lambda = 0.006$, $\lambda = 0.005$, $\lambda = 0.004$, $\lambda = 0.003$, $\lambda = 0.002$, and $\lambda = 0.001$. Table 3 shows the realized performances of models RIML, RIML$^-$, SIML, and EW in terms of the above various performance measures when $\lambda = 0.03$. Table 4 shows the result when $\lambda = 0.02$. Table 5 shows the result when $\lambda = 0.01$. Table 6 shows the result when $\lambda = 0.009$. Table 7 shows the result when $\lambda = 0.008$. Table 8 shows the result when $\lambda = 0.007$. Table 9 shows the result when $\lambda = 0.006$. Table 10 shows the result when $\lambda = 0.005$. Table 11 shows the result when $\lambda = 0.004$. Table 12 shows the result when $\lambda = 0.003$. Table 13 shows the result when $\lambda = 0.002$. Table 14 shows the result when $\lambda = 0.001$. In order to test whether the Sharpe ratio of our model RIML outperforms those of other models significantly, we employ a significance testing method about Sharpe ratios proposed by Jobson and Korkie [21]. The corresponding one-sided p values are presented in the column of the Sharpe ratio. $***$, $**$, and $*$ indicate that the Sharpe ratio of our model RIML outperforms that of the corresponding model significantly at the 1% level, 5% level, and 10% level, respectively. For all cases of the risk aversion coefficient λ, our model RIML consistently performs best in terms of return and risk-adjusted return measures among the four models. The Sharpe ratio of our model RIML is significantly larger than those of the other three models. Interestingly, we find that the realized performances of models RIML$^-$ and SIML are very similar. For all performance measures, our model

Table 3: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.03$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001705	0.059488	0.058076	0.514327	0.037829	0.001223	3.351051
RIML$^-$	0.001370	0.047292** (0.022363)	0.046309	0.509334	0.029937	0.000975	2.504079
SIML	0.001348	0.046351** (0.016264)	0.045297	0.506573	0.029458	0.000949	2.359805
EW	0.000742	0.029613** (0.049575)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 4: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.02$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001726	0.060390	0.059258	0.519263	0.038103	0.001242	3.475284
RIML$^-$	0.001295	0.044441*** (0.004550)	0.043359	0.504967	0.028298	0.000911	2.282259
SIML	0.001290	0.044243*** (0.004131)	0.043196	0.504939	0.028195	0.000907	2.286590
EW	0.000742	0.029613** (0.046820)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 5: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.01$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001703	0.059507	0.058433	0.518954	0.037254	0.001223	3.565365
RIML$^-$	0.001283	0.043964*** (0.004513)	0.042915	0.505353	0.028044	0.000902	2.265623
SIML	0.001286	0.044053*** (0.004818)	0.043006	0.505152	0.028105	0.000903	2.245739
EW	0.000742	0.029613* (0.052438)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 6: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.009$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001699	0.059338	0.058265	0.518775	0.037128	0.001220	3.597911
RIML$^-$	0.001283	0.043950*** (0.004719)	0.042902	0.505258	0.028037	0.000902	2.256801
SIML	0.001280	0.043853*** (0.004732)	0.042812	0.504836	0.027976	0.000899	2.224514
EW	0.000742	0.029613* (0.053487)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 7: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.008$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001693	0.059139	0.058068	0.518561	0.037014	0.001216	3.623607
RIML$^-$	0.001285	0.044023*** (0.005261)	0.042975	0.505304	0.028084	0.000903	2.247054
SIML	0.001271	0.043547*** (0.004598)	0.042515	0.504746	0.027783	0.000893	2.207364
EW	0.000742	0.029613* (0.054733)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 8: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.007$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001688	0.058926	0.057856	0.518395	0.036891	0.001211	3.620472
RIML$^-$	0.001290	0.044194*** (0.006244)	0.043147	0.505530	0.028196	0.000907	2.246750
SIML	0.001281	0.043870*** (0.005724)	0.042836	0.505163	0.027993	0.000900	2.222131
EW	0.000742	0.029613* (0.056102)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 9: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.006$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001683	0.058726	0.057660	0.518251	0.036779	0.001207	3.611440
RIML$^-$	0.001290	0.044200*** (0.006894)	0.043154	0.505624	0.028203	0.000907	2.236750
SIML	0.001287	0.044086*** (0.006774)	0.043048	0.505264	0.028130	0.000904	2.235418
EW	0.000742	0.029613* (0.057409)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 10: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.005$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001678	0.058527	0.057466	0.505743	0.036670	0.001203	3.597434
RIML$^-$	0.001291	0.044209*** (0.007266)	0.043167	0.505743	0.028213	0.000907	2.236180
SIML	0.001295	0.044358*** (0.007821)	0.043314	0.505397	0.028301	0.000910	2.249519
EW	0.000742	0.029613* (0.058741)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 11: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.004$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001672	0.058311	0.057255	0.517977	0.036551	0.001198	3.582173
RIML$^-$	0.001298	0.044450*** (0.008215)	0.043414	0.506065	0.028373	0.000912	2.245132
SIML	0.001318	0.045132** (0.010416)	0.044098	0.506448	0.028813	0.000926	2.281277
EW	0.000742	0.029613* (0.060217)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 12: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.003$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001667	0.058083	0.057032	0.517816	0.036426	0.001194	3.552670
RIML$^-$	0.001318	0.045117** (0.011217)	0.044097	0.506885	0.028816	0.000927	2.273388
SIML	0.001344	0.045975** (0.015369)	0.044981	0.507586	0.029386	0.000945	2.307677
EW	0.000742	0.029613* (0.061814)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 13: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.002$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001661	0.057843	0.056797	0.517629	0.036296	0.001189	3.556490
RIML$^-$	0.001337	0.045719** (0.016084)	0.044725	0.507588	0.029223	0.000940	2.293018
SIML	0.001335	0.045646** (0.016097)	0.044649	0.507032	0.029169	0.000938	2.287590
EW	0.000742	0.029613* (0.063534)	0.027876	0.470055	0.019598	0.000605	1.826263

Table 14: Realized performances of models RIML, RIML$^-$, SIML, and EW according to various performance measures when $\lambda = 0.001$.

	Mean	Sharpe ratio	Downside Sharpe	UP ratio	Mean/VaR	Mean/CVaR	Cumulative return
RIML	0.001654	0.057576	0.056534	0.517410	0.036149	0.001183	3.542119
RIML$^-$	0.001341	0.045846** (0.020987)	0.044856	0.507387	0.029304	0.000942	2.287160
SIML	0.001351	0.046203** (0.024616)	0.045205	0.507646	0.029533	0.000949	2.288830
EW	0.000742	0.029613* (0.065501)	0.027876	0.470055	0.019598	0.000605	1.826263

RIML outperforms models RIML$^-$ and SIML. The experimental result demonstrates the benefit of accounting for moments ambiguity in a robust international portfolio optimization model using first-order LPM as the risk measure. For all cases of the risk aversion coefficient λ, the model EW consistently has the worst performance in terms of return and risk-adjusted return measures. DeMiguel et al. [22] find that, in their numerical experiments, the equally weighted model EW outperforms sample-based mean-variance strategy and various extensions of sample-based mean-variance strategy, which are designed to deal with the problem of estimation error. Hence, the result that our model RIML significantly outperforms the model EW is very encouraging. Our empirical results show that when λ is set small, the variation of λ does not change the performances of models RIML, RIML$^-$, and SIML significantly.

5. Conclusions

In this paper, we propose a robust international portfolio optimization model with worst-case LPM and mean return. In this model, we assume that the distributions and the first- and second-order moments of distributions of future returns of assets and exchange rates are ambiguous. We reformulate the proposed model into an equivalent SDP problem which is computationally tractable. For investigation of the performance of our proposed model, we also give two benchmark models. In the first benchmark model SIML, we use historical returns to form approximations of the distributions of future returns and build a scenario-based international portfolio optimization model under these approximations of distributions. In the second benchmark model RIML$^-$, we assume that the distributions of future

returns are ambiguous, but the first- and second-order moments of distributions are known beforehand. We also reformulate this model into an equivalent SDP problem. We conduct empirical experiments in a rolling forward way using the return measures and various risk-adjusted return measures to compare the out-of-sample performances of the four models RIML, RIML$^-$, SIML, and an equally weighted model EW. The result demonstrates the superiority of our model RIML over other models. It shows that investors can get benefits when accounting for the ambiguity of the first- and second-order moments. It also verifies that robust models outperform scenario-based model and equally weighted model.

Data Availability

The data used to support the findings of this study are available from the database Wind with the web address https://www.wind.com.cn/.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 71720107002 and 71971086), the Joint Foundation of National Natural Science Foundation of China-Guangdong Province (no. U1901223), the Foundation for Key Program of Ministry of Science and Technology of China (no. 2020AAA0108404), the Fundamental Research Funds for the Central Universities (nos. 2018JDXM02 and 2019ZD13), the Financial Service Innovation and Risk Management Research Base of Guangzhou of China, the Guangdong Basic and Applied Basic Research Foundation (no. 2019B151502037), and GDUPS (Liu Yongjun 2019).

References

[1] H. G. Grubel, "Internationally diversified portfolios: welfare gains and capital flows," *The American Economic Review*, vol. 58, pp. 1299–1314, 1968.

[2] H. Levy and M. Sarnat, "International diversification of investment portfolios," *The American Economic Review*, vol. 60, pp. 668–675, 1970.

[3] D. R. Lessard, "International portfolio diversification: a multivariate analysis for a group of Latin american countries," *The Journal of Finance*, vol. 28, no. 3, pp. 619–633, 1973.

[4] N. Coeurdacier and S. Guibaud, "International portfolio diversification is better than you think," *Journal of International Money and Finance*, vol. 30, no. 2, pp. 289–308, 2011.

[5] M. Umutlu and S. G. Yargı, "To diversify or not to diversify internationally?" *Finance Research Letters*, vol. 44, Article ID 102110, 2022.

[6] M. Jasemi, L. Monplaisir, and P. Amini Jam, "Development of an efficient method to approximate the risk measure of lower partial moment of the first order," *Computers & Industrial Engineering*, vol. 135, pp. 326–332, 2019.

[7] V. S. Bawa, "Optimal rules for ordering uncertain prospects," *Journal of Financial Economics*, vol. 2, no. 1, pp. 95–121, 1975.

[8] V. S. Bawa and E. B. Lindenberg, "Capital market equilibrium in a mean-lower partial moment framework," *Journal of Financial Economics*, vol. 5, no. 2, pp. 189–200, 1977.

[9] P. C. Fishburn, "Mean-risk analysis with risk associated with below-target returns," *The American Economic Review*, vol. 67, pp. 116–126, 1977.

[10] L. Chen, S. He, and S. Zhang, "Tight bounds for some risk measures, with applications to robust portfolio selection," *Operations Research*, vol. 59, no. 4, pp. 847–865, 2011.

[11] M. Ç. Pınar and A. B. Paç, "Mean semi-deviation from a target and robust portfolio choice under distribution and mean return ambiguity," *Journal of Computational and Applied Mathematics*, vol. 259, pp. 394–405, 2014.

[12] A. Ling, J. Sun, and X. Yang, "Robust tracking error portfolio selection with worst-case downside risk measures," *Journal of Economic Dynamics and Control*, vol. 39, pp. 178–207, 2014.

[13] A. Ling and L. Tang, "A numerical study for robust active portfolio management with worst-case downside risk measure," *Mathematical Problems in Engineering*, vol. 2014, Article ID 912389, 13 pages, 2014.

[14] A. Ling, J. Sun, and M. Wang, "Robust multi-period portfolio selection based on downside risk with asymmetrically distributed uncertainty set," *European Journal of Operational Research*, vol. 285, no. 1, pp. 81–95, 2020.

[15] E. Delage and Y. Ye, "Distributionally robust optimization under moment uncertainty with application to data-driven problems," *Operations Research*, vol. 58, no. 3, pp. 595–612, 2010.

[16] A. Shapiro, "On duality theory of conic linear problems," in *Semi-infinite Programming*, pp. 135–165, Springer, Berlin, Germany, 2001.

[17] J. Lofberg, "Yalmip: a toolbox for modeling and optimization in matlab," in *Proceedings of the IEEE international conference on robotics and automation (IEEE Cat. No. 04CH37508)*, pp. 284–289, IEEE, Taipei, Taiwan, September 2004.

[18] W. F. Sharpe, "The sharpe ratio," *Journal of Portfolio Management*, vol. 21, no. 1, pp. 49–58, 1994.

[19] W. T. Ziemba, "The symmetric downside-risk sharpe ratio," *Journal of Portfolio Management*, vol. 32, no. 1, pp. 108–122, 2005.

[20] F. A. Sortino and R. V. D. Meer, "Downside risk," *Journal of Portfolio Management*, vol. 17, no. 4, pp. 27–31, 1991.

[21] J. D. Jobson and B. M. Korkie, "Performance hypothesis testing with the sharpe and treynor measures," *The Journal of Finance*, vol. 36, no. 4, pp. 889–908, 1981.

[22] V. DeMiguel, L. Garlappi, and R. Uppal, "Optimal versus naive diversification: how inefficient is the 1/NPortfolio strategy?" *Review of Financial Studies*, vol. 22, no. 5, pp. 1915–1953, 2009.

Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 5617213, 7 pages
https://doi.org/10.1155/2022/5617213

Research Article

A Smoothing SAA Method for Solving a Nonconvex Multisource Supply Chain Stochastic Optimization Model

Chunlin Deng,[1] Yao Xiong,[2] Liu Yang,[3] and Yi Yang[4]

[1]*School of Public Administration, Xiangtan University, Xiangtan 411 105, Hunan, China*
[2]*School of Mathematics and Computational Sciences Xiangtan University, Xiangtan 411 105, Hunan, China*
[3]*Hunan Key Laboratory for Computation and Simulation, Science and Engineering, School of Mathematics and Computational Science, Xiangtan University, Xiangtan 411 105, Hunan, China*
[4]*School of Mathematics and Computational Sciences Xiangtan University, Xiangtan 411 105, Hunan, China*

Correspondence should be addressed to Liu Yang; yangl410@163.com

Received 30 October 2021; Revised 28 December 2021; Accepted 19 January 2022; Published 22 February 2022

Academic Editor: Guoqiang Wang

We construct a new multisource supply chain stochastic optimization model when the supply and demand are both uncertain. This model is nonconvex because the decision variables are truncated by the random variable in the objective function. It is a common technical challenge encountered in many operations management models. To address this challenge, we adopt a novel transformation technique to transform the nonconvex problem into an equivalent convex optimization problem. Then, we provide a smoothing sample average approximation (SAA) method to solve the transformed problem. The SAA model is a good approximation for the expected value function in the objective function when the number of samples is large enough. The smoothing technique can transfer the nonsmooth plus function into a smoothing function in the model, and thus, we can use the numerical methods for the common nonlinear integer programming to solve the transformed model. Numerical tests verify the effectiveness of the new model and the smoothing SAA method.

1. Introduction

In the upstream structure of a multisupplier-single-factory supply chain, multisource decision-making is considered in the environment of uncertain demand and supply. In order to deal with the risk of these supply uncertainties, most companies adopt multiple supply source strategies, i e., when there is only one major partner supplier; in order to reduce the risk of supply uncertainty caused by emergencies, most enterprises will choose another supply source as a backup supplier [1–5]. For example, in the aforementioned Philips interruption of the provision of radio frequency chip events, Nokia's backup quickly responded and quickly put into production, but Ericsson eventually exited the mobile phone business market because of the lack of backup suppliers.

The multisource supply chain stochastic optimization model was first studied by Barankin [6] in the one-period setting and then extended by Daniel [7], Fukuda [8], and Whittmore and Saunders [9] to the various settings, see [6–9]. In 2012, Feng and Shi considered a joint inventory control and pricing problem with multiple suppliers whose supply capacities are uncertain [10]. They show that, with deterministic capacities, ordering from a cheaper source first is optimal. However, when the supply capacities are random, such a policy is not optimal. They show that the optimal policy can be characterized by a near reorder point. In 2014, Zhou and Chao [11] provide a dual-sourcing problem with price sensitive demand and a regular supplier; they characterize the structure of the optimal policy of the dual-sourcing problem. Then, Gong et al. [12] generalized the structural analysis to a dual-sourcing problem with price sensitive demand and Markovian supply interruptions. In all these models, there are no capacity limits on the supplies. To the best of our knowledge, most current research work simply assumed that supply accounts for a certain proportion of orders, ignoring supply uncertainties, i.e., uncertainties arising from unreliable

product supply processing processes, such as suppliers delivering only part of the product or canceling orders [1–12]. Each potential supplier corresponded to a random supply and a unit ordering cost, the enterprise must make a decision on the quantity ordered by each supplier before the demand and supply was realized, so the final volume of transportation was the minimum between supply and order quantity. So, in this paper, we introduce the decision variable truncated by random variables and construct a multisource supply chain optimization model which is more in line with the actual construction of the supply uncertainty environment. Because the random variable truncation destroys the convexity of the problem, it is very challenging to solve and analyze this kind of nonconvex optimization problem. Based on the work of [4, 5], we first transform this problem into an equivalent convex minimum problem and then give a smoothing SAA algorithm to solve the problem. Finally, the validity of the model and algorithm is verified by numerical examples.

The structure of this paper is as follows. In Section 2, we establish a stochastic optimization model for the multisource supply chain problem with decision variables truncated by random variables. In Section 3, a novel transformation technique is used to transform the nonconvex multisource supply chain optimization model into an equivalent convex optimization model. In Section 4, for the solution of the transformed model, the expected value function is firstly treated with the sample average approximation method, and then, a smoothing algorithm is provided to deal with the nonsmoothness in the model. In Section 5, we provide a numerical test for a specific dual-source supply chain problem and analyze the impact of the ordering cost, supply capacity, and other factors on the enterprise's ordering strategy. The conclusion of this paper is given in Section 6.

2. Establishment of the Model

We consider a single cycle and single product supply chain with multiple suppliers $\mathcal{N} = \{1, \ldots, n\}$ in uncertain demand and supply environment. It is assumed that there is no fixed cost, and the quantity supplied by supplier $j \in \mathcal{N}$ is K_j. Here, we assume that the sequence $\{K_j\}$, $j \in \mathcal{N}$, is independent of each other, and they are random variables of the same distribution. Demand d is also random, and the random distribution of supply quantity K_j is independent of each other. The unit ordering cost at the supplier is c_j; without loss of generality, we assume that $c_1 \le c_2 \le \cdots \le c_n$. The event is divided into two phases: the ordering phase and the selling phase. In the ordering stage, the enterprise first reviews the existing inventory level and the unprocessed orders and sets the order before observing the supplier's supply quantity K_j. Let q_j represent the ordering quantity at supplier $j \in \mathcal{N}$, and we use k_j to represent the realized supply quantity of K_j. Then, the actual supply obtained by the enterprise is $q_j \wedge k_j$, where $\cdot\wedge\cdot = \min(\cdot,\cdot)$ is the minimum component symbol (see the same definition in [4]). Noting that the supply uncertainty of the supplier has been solved at this time, the same settings were set in the study on the stochastic production problem in [13] because the supply uncertainty at this time is mainly due to the unreliable production process and production time. In the selling phase, the requirements are realized, the remaining products are processed, and the unmet requirements are backlogged. The unit shortage cost and the unit residual value are h^- and h^+, respectively. Let p represent the unit retail price of the product. The enterprise's goal is to develop a multisource ordering strategy to minimize the total expected cost. Based on random variable truncation, we establish the following unconstrained multisource supply chain optimization model:

$$\min E[f(q\wedge k)] = E\left[\sum_{j\in\mathcal{N}} c_j\left(q_j\wedge k_j\right) + h^-\left(d - \sum_{j\in\mathcal{N}}\left(q_j\wedge k_j\right)\right)_+ - p\left(d - \left(d - \sum_{j\in\mathcal{N}}\left(q_j\wedge k_j\right)\right)_+\right) - h^+\left(\sum_{j\in\mathcal{N}}\left(q_j\wedge k_j\right) - d\right)_+\right], \tag{1}$$

where $(x)_+ = \max(x, 0)$, $f\colon \mathcal{F}^n \longrightarrow \overline{\mathbb{R}}$ is the function of the decision variable q_j, here $\mathcal{F}$ is either the real space or the space with integers $\mathcal{Z}$ and $\overline{\mathbb{R}} = \mathbb{R} \cup +\infty$, k_j, d are random variables, and the support sets of the random vectors $k = (k_1, k_2, \ldots, k_n)$ and $d = (d_1, d_2, \ldots, d_n)$ are $\mathrm{Supp}(k) = \mathcal{X} \subset \mathcal{F}^n$ and $\mathrm{Supp}(d) = \mathcal{Y} \subset \mathcal{F}$, respectively.

In model (1), the first and second items in the objective function represent the total order cost and the total shortage cost, respectively, and the third and fourth items represent the total profit on the sale and the residual value, respectively. As a matter of convenience, model (1) can be arranged as follows:

$$\min E[f(q\wedge k)] = E\left[\sum_{j\in\mathcal{N}} c_j\left(q_j\wedge k_j\right) + \left(h^- + p\right)\left(d - \sum_{j\in\mathcal{N}}\left(q_j\wedge k_j\right)\right)_+ - h^+\left(\sum_{j\in\mathcal{N}}\left(q_j\wedge k_j\right) - d\right)_+ - p\,d\right]. \tag{2}$$

3. Equivalent Transformation Technique

In general, even if f is convex, problem (2) is a nonconvex optimization problem due to the existence of truncated item $(q_j\wedge k_j)$. As Ciarallo et al. pointed out in [14], for the production planning problem of single product and single cycle supply chain with uncertain supply and demand, random supply will lead to the generation of a single peak

and nonconvex objective function, and they also verified that the objective expectation function has quasi-convexity.

Recently, Chen Xin et al. [4, 5] proposed an effective conversion technique to transform a nonconvex minimization problem into a convex minimization problem and further showed that this method could maintain some good structural properties, such as convexity, submodality, and $L^{\natural}$– convexity. Therefore, based on the result in [4, 5], we get the following theorem for problem (1).

Theorem 1. *Assume that (a) function $f\colon \mathscr{F}^n \longrightarrow \overline{\mathfrak{R}}$ is a lower semicontinuous function and satisfies $f(x) \longrightarrow \infty$, $|x| \longrightarrow \infty$; (b) f is the component convex function (if $\mathscr{F} = \mathscr{Z}$, f is the component discrete convex function); (c) the components of the random vector k are independent of each other, and the corresponding realization is $k_j \in \mathscr{X} = Supp(k)$. Let ϑ^* represent the optimal value of model (2); then, ϑ^* is also the optimal objective value of the following optimization model:*

$$\begin{aligned} \min, & E[f(v(k))] = E\left[\sum_{j\in\mathscr{N}} c_j \cdot v_j(k_j) + (h^- + p)\left(d - \sum_{j\in\mathscr{N}} v_j(k_j)\right)_+ - h^+\left(\sum_{j\in\mathscr{N}} v_j(k_j) - d\right)_+ - p\ d\right], \\ \text{s.t.,} & v_j(k_j) = q_j \wedge k_j, v(k) \\ & \quad = (v_1(k_1), \ldots, v_n(k_n)). \end{aligned} \tag{3}$$

Proof. Let σ^* be the optimal target value of model (3), since, for any $q \in \mathscr{F}^n$, the equality constraint $v_j(k_j) = q_j \wedge k_j$ is feasible in model (3), so we get $\sigma^* \le \vartheta^*$.

Next, we just need to prove that $\vartheta^* \le \sigma^*$. Obviously, the conclusion is right when $\sigma^* = \infty$, so in the following analysis, we firstly assume $\sigma^* < \infty$; then, combining condition (a), we can know that models (2) and (3) both have finite optimal solutions. In the following, we can prove that given any optimal solution $v^* = \{v(k_j) | k_j \in \mathscr{X}\}$, we can find a solution $\widehat{q} \in \mathscr{F}^n$ that satisfies $E[f(\widehat{q} \wedge k)] = E[f(v^*(k))]$ by mathematical induction.

When $n = 1$, let $\widehat{q} = \operatorname{argmin}_{q\in\mathscr{F}^n} f(q)$ (take the minimum value if there are multiple optimal solutions), and for any feasible solution of model (3), according to Lemma 3.1 in [4], we have $f(\widehat{q} \wedge k_j \le f(v(k_j)))$, for $\forall k_j \in \mathscr{X}$, so we get $E[f(\widehat{q} \wedge k)] \le \sigma^*$. Note that $\widehat{q}$ is a feasible solution of model (2), which means that $\vartheta^* = E[f(\widehat{q} \wedge k)] \le \sigma^*$; combined with $\sigma^* \le \vartheta^*$, we get $\sigma^* = \vartheta^*$.

When $n > 1$, let q_j^*, $j = 1, \ldots, n$, be the jth element of q^*; starting with the first element, we define

$$\sigma_1(q_1) = E[f(q_1, v_2^*(k_2), \ldots, v_n^*(k_n))]. \tag{4}$$

In condition (b), the component convexity of f means that $\sigma_1(q_1)$ is convexity on q_1. Since all components of the vector k obey a relatively independent and identical distribution,
$E_{k_1}[\sigma_1(v_1^*(k_1))] = E_{k_1}[f(v_1(k_1)), v_2^*(k_2), \ldots, v_n^*(k_n)])$ is valid for any measurable function $v_1(k_1)$, and based on the previous analysis of case $n = 1$, we have that there is a $\widehat{q}_1$, which satisfies the following equation:

$$\begin{aligned} \sigma^* &= \min\{E[\sigma_1(v_1(k_1))] | v_1(k_1) \le k_1, \quad \forall k_1 \\ &\quad \in \mathscr{X}_1, v_1(k_1) \in \mathscr{F}\} \\ &= \min_{q_1\in\mathscr{F}} E[\sigma_1(q_1 \wedge k_1)] \\ &= E[\sigma_1(\widehat{q}_1 \wedge k_1)]. \end{aligned} \tag{5}$$

Then, we go on to define $\sigma_2(q_2) = E[f(\widehat{q}_1 \wedge k_1, q_2, v_3^*(k_3), \ldots, v_n^*(k_n))]$, and obviously, σ_2 is also convex; similarly, there is $\widehat{q}_2$ that satisfies the following equation:

$$\begin{aligned} \sigma^* &= \min\{E[\sigma_2(v_2(k_2))] | v_2(k_2) \le k_2, \quad \forall k_2 \in \mathscr{X}_2, v_2(k_2) \in \mathscr{F}\} \\ &= \min_{q_2\in\mathscr{F}} E[\sigma_2(q_2 \wedge k_2)] \\ &= E[\sigma_2(\widehat{q}_2 \wedge k_2)]. \end{aligned} \tag{6}$$

Repeat the above steps, and define $\sigma_i(q_i) = E[f(\widehat{q}_1 \wedge k_1, \ldots, \widehat{q}_{i-1} \wedge k_{i-1}, q_i,$
$v_{i+1}^*(k_{i+1}), \ldots, v_n^*(k_n))]$. In the same way, we can find $\widehat{q}_i, i = 3, \ldots, n$, which satisfies the following equations:

$$\begin{aligned} \sigma^* &= \min\{E[\sigma_i(v_i(k_i))] | v_i(k_i) \le k_i, \forall k_i \in \mathscr{X}_i, v_i(k_i) \in \mathscr{F}\} \\ &= \min_{q_i\in\mathscr{F}} E[\sigma_i(q_i \wedge k_i)] \\ &= E[\sigma_i(\widehat{q}_i \wedge k_i)]. \end{aligned} \tag{7}$$

So,

$$\begin{aligned} \sigma^* &= E[\sigma_n(\widehat{q}_n \wedge k_n)] \\ &= E[f(\widehat{q}_1 \wedge k_1, \ldots, \widehat{q}_n \wedge k_n)]. \end{aligned} \tag{8}$$

Since $\widehat{q}$ is a feasible solution of model (3), we know that $\vartheta^* \le E[f(\widehat{q} \wedge k)] = \sigma^*$; combined with $\sigma^* \le \vartheta^*$, we get $\sigma^* = \vartheta^*$.

Based on the above theorem, we successfully transformed the nonconvex supply chain optimization model (2) into an equivalent convex supply chain optimization model (3), see [4, 5]. □

4. Solution Method of Model (3)

To solve model (3), we firstly use the SAA method to approximate the expected value function in the objective function. It is well known that Shapiro have proved that, under some regularization conditions, the optimal value of SAA problem converges to the optimal value of the original stochastic programming problem according to probability 1

as the number of samples approaches infinity (Chapter 6 of [15]). Assume that $k_j^m, m = 1, \ldots, M$, and $d^l, l = 1, \ldots, L$, are the random samples generated by supply k_j, $j = 1, \ldots, n$, and demand d, respectively; then, the SAA model for multisource supply chain problems is

$$\begin{aligned} &\min \frac{1}{M}\sum_{m=1}^{M}\sum_{j\in\mathcal{N}} c_j \cdot v_j\left(k_j^m\right) + \frac{1}{ML}\sum_{m=1}^{M}\sum_{l=1}^{L}\left(h^- + p\right)\cdot\left(d^l - \sum_{j\in\mathcal{N}} v_j\left(k_j^m\right)\right)_+ - \frac{1}{ML}\sum_{m=1}^{M}\sum_{l=1}^{L} h^+ \cdot \left(\sum_{j\in\mathcal{N}} v_j\left(k_j^m\right) - d^l\right)_+ - \frac{1}{L}\sum_{l=1}^{L} p\cdot d^l \\ &v_j\left(k_j^m\right) \le k_j^m, j = 1, \ldots, n, v\left(k^m\right) \\ &\text{s.t. } = \left(v_1\left(k_1^m\right), \ldots, v_n\left(k_n^m\right)\right). \end{aligned} \tag{9}$$

Obviously, there is a nonsmooth plus function $(\cdot)_+$ in model (9) which will cause difficulty in computing this problem. So, we use a smoothing approach in [16–19] to deal with nonsmoothness in multisource supply chain problems. For the sake of simplicity, we denote

$$G(v(k)) = \frac{1}{M}\sum_{m=1}^{M}\sum_{j\in\mathcal{N}} c_j \cdot v_j\left(k_j^m\right) + \frac{1}{\mathrm{ML}}\sum_{m=1}^{M}\sum_{l=1}^{L}\left(h^- + p\right)\cdot\left(d^l - \sum_{j\in\mathcal{N}} v_j\left(k_j^m\right)\right)_+ - \frac{1}{\mathrm{ML}}\sum_{m=1}^{M}\sum_{l=1}^{L} h^+ \cdot \left(\sum_{j\in\mathcal{N}} v_j\left(k_j^m\right) - d^l\right)_+ - \frac{1}{L}\sum_{l=1}^{L} p\cdot d^l. \tag{10}$$

Let $t > 0$ be a smooth parameter; we construct the following smoothing approximation functions by using the same technique in [14]:

$$\begin{cases} g_t(t, v(k)) = t\,\ln\left[1 + \exp\left(\dfrac{d^l - \sum_{j\in\mathcal{N}} v_j\left(k_j^m\right)}{t}\right)\right], \\ h_t(t, v(k)) = t\,\ln\left[1 + \exp\left(\dfrac{d^l - \sum_{j\in\mathcal{N}} v_j\left(k_j^m\right) - d^l}{t}\right)\right], \end{cases} \tag{11}$$

$$\overline{G}(t, v(k)), = \frac{1}{M}\sum_{m=1}^{M}\sum_{j\in\mathcal{N}} c_j \cdot v_j\left(k_j^m\right) + \frac{1}{\mathrm{ML}}\sum_{m=1}^{M}\sum_{l=1}^{L}\left(h^- + p\right)\cdot g_t(t, v(k)) - \frac{1}{\mathrm{ML}}\sum_{m=1}^{M}\sum_{l=1}^{L} h^+ \cdot h_t(t, v(k)) - \frac{1}{L}\sum_{l=1}^{L} p\cdot d^l,$$

where $g_t(t, v(k))$, $h_t(t, v(k))$, and $\overline{G}(t, v(k))$ are the smooth approximation functions of $g(v(k)), h(v(k)), G(v(k))$, respectively. We can prove that the smooth functions have the following properties.

Theorem 1. *For $\forall t > 0$, we can obtain*

(i) $\overline{G}(t, v(k))$ is an increasing function of t, and we have

$$\left\|\overline{G}(t, v(k)) - G(v(k))\right\| \le t\,\ln t. \tag{12}$$

(ii) $G(v(k))$ is a convex function of $v(k)$, and $\overline{G}(t, v(k))$ remains convex of $G(v(k))$.

(iii) $\overline{G}(t, v(k))$ is a continuous differentiable function whose first derivative is

$$\nabla_{v(k)}\overline{G}(t, v(k)) = \frac{1}{\mathrm{ML}}\sum_{m=1}^{M}\sum_{l=1}^{L}\left[\left(h^- + p\right)\cdot\frac{\exp\left(\left(d^l - \sum_{j\in\mathcal{N}} v_j\left(k_j^m\right)/t\right)\right)}{1 + \exp\left(d^l - \sum_{j\in\mathcal{N}} v_j\left(k_j^m\right)/t\right)} - h^+ \cdot \frac{\exp\left(d^l - \sum_{j\in\mathcal{N}} v_j\left(k_j^m\right) - d^l/t\right)}{1 + \exp\left(\sum_{j\in\mathcal{N}} v_j\left(k_j^m\right) - d^l/t\right)} + \sum_{j\in\mathcal{N}} c_j\right]. \tag{13}$$

(iv) For any fixed $v(k) \in \mathfrak{R}^{\mathcal{N}\times M}$, there is

$$\left\|\nabla_{v(k)}\overline{G}(t, v(k)) - \partial G(v(k))\right\| = o(t). \tag{14}$$

Proof. According to Lemma 3.1 in [16], conclusion (*i*), (*iii*), and (*iv*) are obviously valid. For conclusion (*ii*), since $\overline{G}(t, v(k))$ is the sum function of linear functions $g_t(t, v(k))$, $h_t(t, v(k))$, and $g_t(t, v(k))$ and $h_t(t, v(k))$ are convex functions and the sum function of convex functions remains convex, conclusion (*ii*) is valid.

According to Theorem 1, we know that the smooth functions $g_t(t, v(k))$, $h_t(t, v(k))$, and $\overline{G}(t, v(k))$ are all convex functions, and when the smooth factor $t \longrightarrow 0^+$, the smooth functions have a good approximation effect. Therefore, we can construct the following smoothing model of multisource supply chain to solve model (2):

$$\begin{aligned} &\min \overline{G}(t, v(k)) \\ &\text{s.t. } v_j\left(k_j^m\right) \le k_j^m, \quad j = 1, \ldots, n, v\left(k^m\right) \\ &\quad = \left(v_1\left(k_1^m\right), \ldots, v_n\left(k_n^m\right)\right). \end{aligned} \tag{15}$$

Model (15) is a smooth nonlinear convex programming problem with dimension $n \times M$ (n is the number of suppliers). □

5. Numerical Tests

In this section, we give numerical tests for the new model (1) and the smoothing SAA method. The computation is performed in MATLAB R2021b, in a computer with CPU Apple M1 and RAM 16 GB.

Consider a double source supply chain problems when supply and demand are both uncertain. The experimental data are from [1] in which the enterprise has two suppliers: the unit ordering costs of each supplier are c_1 and c_2, respectively; the unit shortage cost and the rest of the residual value are h^- and h^+, respectively; the retail price of unit product is p. Assume that the market demand of the product d is uniformly distributed on (a, b). In order to describe the positive dependence between random supplies, we consider the case where suppliers share a "market risk." For example, in reality, suppliers suddenly receive urgent orders from enterprises, and the supply of each supplier j is set as $k_j = y_j + z$, $j \in \mathcal{N}$, where y_j and z are mutually independent random variables subject to uniform distribution, the market risk factor z obeys the uniform distribution on (c, d) and is embedded into the supply provided by each supplier, and y_j obeys the uniform distribution on (e_j, f_j).

Set the number of samples $M = L = 100$, the smooth parameter $t = 0.001$, and the basic parameters are set as $c_1 = 50$, $c_2 = 80$, $h^- = 30$, $h_+ = 20$, $p = 100$, $a = 5000$, $b = 10000$, $c = 2500$, and $d = 5000$, when the supply capacity of the two suppliers is the same, that is, $e_1 = e_2 = 2500$ and $f_1 = f_2 = 5000$. Based on the above initial number, we solve the smoothing model (15) and get the optimal ordering strategy $q^* = [7643, 2188]^T$, and the expected profit is $-E[\cdot] = 419300$. The results show that when the supply capacity of the cooperative suppliers is the same, the enterprise gives priority to place orders to supplier 1, that is, the enterprise chooses the supply source based on the ordering cost, which is consistent with the data results in [1].

Then, we consider the influence of smooth parameter t in the algorithm. Set the number of samples $M = L = 100$; the results are shown in Figure 1. It can be seen that the optimal target value does not change significantly. This indicates that the smoothing algorithm is not sensitive to the parameter t and further indicates that the original model is equivalent to the smoothing model when $t \longrightarrow 0^+$.

In the following, we consider the influence of supply capacity on decision-making. Let the sample size $M = L = 100$ and smooth parameter $t = 0.001$; the results are given in Table 1 and Table 2.

From Table 1 and Table 2, we can see that the optimal order quantity of an enterprise is determined by the wholesale price and supply capacity of the supplier. As we can see in Table 1, in the case of $c_1 < c_2$, when the supply capacity of the fixed supplier 1 is $y_1 \sim U(2500, 5000)$, no matter how good or bad the supply capacity of supplier 2 is, the enterprise will choose to place an order with supplier 1 firstly; this may be due to the high supply capacity of supplier 1, and it can make the market demand reach the saturation state, so the enterprise will tend to choose the supplier with the lower order cost. From Table 2, we can see that, although $c_1 < c_2$, the optimal ordering strategy is $q_2^* > q_1^*$ when the supply capacity of supplier 1 is much lower than that of supplier 2. Of course, when the supply capacity of supplier 1 increases, so does its order quantity q_1^*. This indicates that when the market demand does not reach the saturation state, the ordering decision q^* of the enterprise is influenced by the ordering cost c_j and supply capacity k_j of the supplier. Moreover, it seems that, in order to meet the demand of the market, the enterprise will even ignore the loss caused by higher costs and choose suppliers with higher supply capacity.

6. Summary

In this paper, we establish a multisource supply chain optimization model with random variable truncation to deal with the uncertainty of the supply and demand. By introducing a new variable, we transfer the nonconvex multisource supply chain model into a convex problem; then, we provide a smoothing SAA algorithm to solve the equivalent problem. The equivalence of the transformed models and the convergence properties of the smoothing approximation function are also given, and numerical tests show that (i) the smoothing SAA method can solve the multisource supply chain optimization model efficiently, (ii) when the supply capacity of the supplier is consistent, the enterprise will choose the supply source based on the ordering cost of the supplier, and (iii) the ordering decision of enterprises is influenced by both ordering cost and supply capacity. In other words, when the supply capacity of suppliers with low ordering cost is large enough to make the market demand reach saturation state, enterprises tend to choose suppliers with low ordering cost. When the supply capacity of suppliers with lower ordering costs cannot meet the market demand, enterprises tend to trust suppliers with higher supply capacity.

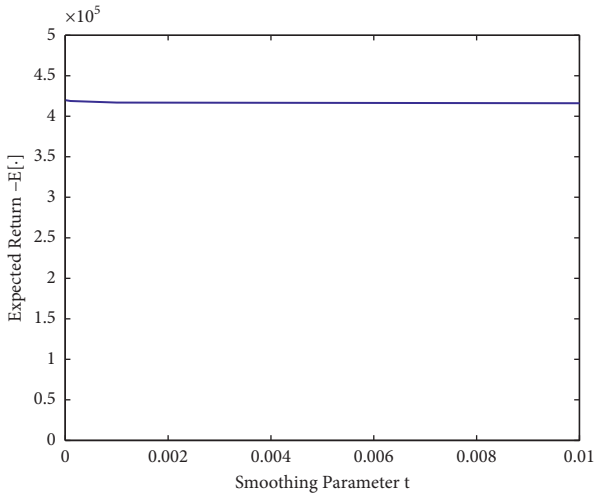


Figure 1: The effect of the smoothing parameter on the object function.

Table 1: The influence of supply capacity on enterprise decision-making when $y_1 \sim U(2500, 5000)$ and $y_2 \sim U(0, 2000)$.

e_1	2500	2500	2500	2500	2500
f_1	5000	5000	5000	5000	5000
e_2	0	2000	4000	6000	8000
f_2	2000	4000	6000	8000	10 000
q_1^*	7705	7531	7576	7510	7531
q_2^*	1920	2354	2228	2353	2354
$-E[\cdot]$	416 850	417 160	413 940	416 000	417 160
CPU (s)	5.486 978	5.609 130	5.564 075	5.536 381	5.552 392

Table 2: The influence of supply capacity on enterprise decision-making when $y_1 \sim U(1000, 5000)$ and $y_2 \sim U(5000, 9000)$

e_1	0	1000	2000	3000	4000
f_1	1000	2000	3000	4000	5000
e_2	5000	4000	3000	2000	1000
f_2	9000	8000	7000	6000	5000
q_1^*	1471	2476	3983	5509	7244
q_2^*	8447	7547	6075	4546	2779
$-E[\cdot]$	234 930	265 720	310 190	356 860	404 470
CPU (s)	4.805 859	4.770 569	4.796 633	4.816 009	4.751 742

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by Natural Science Foundation of China (12 071 399), Key Projects of Hunan Provincial Education Department (18A048), Project of Scientific Research Fund of Hunan Provincial Science and Technology Department (2018WK4006), and Project of Hunan National Center for Applied Mathematics (2020ZYT003).

References

[1] B. Xu, *Study on Retailer's Ordering and Supplier Input Strategy under Unreliable Supply Conditions*, University of Electronic Science and Technology, Chengdu, China, 2010.

[2] A. Federgruen and N. Yang, "Technical note-procurement strategies with unreliable suppliers," *Operations Research*, vol. 59, no. 4, pp. 1033–1039, 2011.

[3] W. Chen, Q. Feng, and S. Seshadri, "Sourcing from suppliers with random yield for price-dependent demand," *Annals of Operations Research*, vol. 208, no. 1, pp. 557–579, 2013.

[4] X. Chen, X. Gao, and Z. Pang, "Preservation of structural properties in optimization with decisions truncated by random variables and its applications," *Operations Research*, vol. 66, no. 2, pp. 340–357, 2018.

[5] X. Chen and X. Gao, "Technical note-stochastic optimization with decisions truncated by positively dependent random variables," *Operations Research*, vol. 67, no. 5, pp. 1321–1327, 2019.

[6] E. W. Barankin, "A delivery-lag inventory model with an emergency provision (the one-period case)," *Naval Research Logistics Quarterly*, vol. 8, no. 3, pp. 258–311, 1961.

[7] K. Daniel, "A delivery-lag inventory model with emergency," in *Multistage Inventory Models and Techniques*, H. Scarf, D. Gilford, and M. Shelly, Eds., pp. 32–46, Stanford University Press, Stanford, CA, USA, 1963.

[8] Y. Fukuda, "Optimal policies for the inventory problem with negotiable leadtime," *Management Science*, vol. 10, no. 4, pp. 690–708, 1964.

[9] A. S. Whittmore and S. C. Saunders, "Optimal inventory under stochastic demand with two supply options," *SIAM Journal on Applied Mathematics*, vol. 32, no. 2, pp. 293–305, 1977.

[10] Q. Feng and R. Shi, "Sourcing from multiple suppliers for price-dependent demands," *Production and Operations Management*, vol. 21, no. 3, pp. 547–563, 2012.

[11] S. X. Zhou and X. Chao, "Dynamic pricing and inventory management with regular and expedited supplies," *Production and Operations Management*, vol. 23, no. 1, pp. 65–80, 2014.

[12] X. Gong, X. Chao, and S. Zheng, "Dynamic pricing and inventory management with dual suppliers of different lead times and disruption risks," *Production and Operations Management*, vol. 23, no. 12, pp. 2058–2074, 2014.

[13] K. B. Hendricks and V. R. Singhal, "The effect of supply chain glitches on shareholder wealth," *Journal of Operations Management*, vol. 21, no. 5, pp. 501–522, 2003.

[14] F. W. Ciarallo, R. Akella, and T. E. Morton, "A periodic review, production planning model with uncertain capacity and uncertain demand-optimality of extended myopic policies," *Management Science*, vol. 40, no. 3, pp. 320–332, 1994.

[15] A. Shapiro, "Monte Carlo sampling methods," *Handbooks in Operations Research and Management Science*, vol. 10, pp. 353–425, 2003.

[16] L. Yang, Y. Chen, and X. Tong, "Smoothing Newton-like method for the solution of nonlinear systems of equalities and inequalities," *Numerical Mathematics: Theory, Methods and Applications*, vol. 2, no. 2, pp. 224–236, 2009.

[17] X. Tong, L. Qi, F. Wu, and H. Zhou, "A smoothing method for solving portfolio optimization with CVaR and applications in

allocation of generation asset," *Applied Mathematics and Computation*, vol. 216, no. 6, pp. 1723–1740, 2010.

[18] L. Yang, Y. Xiong, and X.-J. Tong, "A smoothing algorithm for a new two-stage stochastic model of supply chain based on sample average approximation," *Mathematical Problems in Engineering*, vol. 2017, no. 10, 7 pages, Article ID 5681502, 2017.

[19] L. Yang, X. Tong, Y. Xiong, and F. Shen, "A smoothing saa algorithm for a portfolio choice model based on second-order stochastic dominance measures," *Journal of Industrial and Management Optimization*, vol. 13, no. 5, pp. 1–15, 2017.

Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 2045630, 11 pages
https://doi.org/10.1155/2022/2045630

Research Article

Auction-Based Capacity Allocation in Two Parallel Machines with Inclusive Processing Set Restrictions

Qianqian Zhu and Xiuli Wang

School of Economics and Management, Nanjing University of Science and Technology, Nanjing 210094, China

Correspondence should be addressed to Xiuli Wang; wangdu0816@163.com

Received 2 December 2021; Revised 24 December 2021; Accepted 5 January 2022; Published 28 January 2022

Academic Editor: Guoqiang Wang

We consider the scarce capacity allocation problem in two parallel machines with inclusive processing set restrictions. Our focus is to design an auction mechanism to allocate the production capacity among several selfish customers effectively and efficiently. In our iterative ascending auction mechanism, we need to design jointly two things: who wins what and who pays what. First, we propose an adaptive ascending pricing policy to ensure that bidding by truthful processing requirements and keeping on bidding until the ask prices reach his real revenue is a dominant strategy for each customer. Second, we establish a facility owner's profit maximization model to determine the capacity allocation in two parallel machines with inclusive processing set restrictions; it is an NP hard problem; we also develop a heuristic based on the Lagrangian relaxation technology to obtain the suboptimal solutions. Our computational analysis shows that the auction mechanism can achieve more than 87% of the global system value.

1. Introduction

In current customer-oriented market, the decentralized operating mode has become very popular in the manufacturing industries. This mode can provide services such as product design, manufacturing, and testing for small and medium-sized enterprises by the establishment of a service platform, which is a good solution to the problem of the insufficient funds and talents for small- and medium-sized enterprises, such as the United Microelectronics Corporation (UMC) which uses the online customer information portal MyUMC to provide transparent processing pricing, processing policies, and real-time processing capacity information to customers and allows the selfish customers to make decentralized decisions on booking processing capacity. However, due to the customers' selfish behaviours, the efficiency of the overall system would be reduced to some extent. Therefore, it is important to design an appropriate mechanism to guide individual competition results towards global optimization for the scarce capacity allocation problem in the decentralized operating mode.

To the best of our knowledge, the auction-based method was a suitable approach to solve the production capacity allocation problem, and it has been applied in a single machine and job shop environment [1, 2]. In practice, there existed a scenario that the facility owner possesses parallel machines and they may have the same speed but differ in their functionality, which could be called inclusive processing set (IPS) restrictions. Our study is focused on this case.

There are many applications for capacity allocation problem with IPS restriction. A classical application is assigning several cloud computing tasks with different memory requirements to processors with the same speed but different memory capacities. Each task can be processed by a processor with a memory capacity no less than its memory requirement. Another application exists in testing enterprises. Assuming there are several customers, each with a set of workpieces that needs to be tested, the testing enterprise has two pieces of equipment with the same speed but they differ in test accuracy. High precision testing equipment can provide high and low precision testing services, but low precision testing equipment can only provide low precision

testing services. Here, the testing organization can be described as the "facility owner," the testing equipment can be described as the "machine," the set of workpieces can be described as the "order," and the problem that the facility owner needs to do is allocating the parallel machine capacity to several competing customers, so as to optimize one or more objectives.

In this study, we propose an iterative ascending auction mechanism to solve the capacity allocation problem in two parallel machines with IPS restrictions. The main research work is as follows. First, we design the auction mechanism that the selfish customers can use private information to make effective decisions without knowing the information and strategies of other customers. The pricing policy can make the customers bid by their truthful processing requirements and keep on bidding until the ask prices reach their real revenue. In this way, the customers' local decisions can prompt the facility owner to make decisions that promote the achievement of the collective goal. Second, the winner determination problem is a bid selecting and scheduling problem in two parallel machines with IPS restrictions, and it is NP hard; we develop a heuristic using the Lagrangian relaxation technique to obtain approximate optimal solutions for the large-sized instances in a reasonable time.

The remaining part of the paper is structured as follows. Section 2 reviews the related work. Then, in Section 3, a detailed description of our problem setting is described. In Section 4, a global optimization problem is studied for mainly benchmarking purposes. In Section 5, an auction mechanism is proposed for the capacity allocation in two parallel machines with IPS restrictions. The pricing and winner determination problems, as the main components of our auction mechanism, are discussed in Sections 5.2 and 5.3, respectively. In Section 6, a computational analysis of the performance for the auction mechanism is presented. Finally, Section 7 provides concluding remarks.

2. Literature Review

In this section, we provide a brief review of the relevant literature in the fields of decentralized scheduling, auctions for resource allocation problems, and parallel machine scheduling with IPS restrictions.

2.1. Decentralized Scheduling. In a decentralized scheduling problem, the facility owner and several customers make decisions rationally for their own interest, and their selfish behaviours result in a situation that can be characterized as system equilibrium. However, such a system equilibrium may lead to suboptimal system performance from a global perspective. The literature on this subject mainly has the following two perspectives: (a) assuming all information is public, characterize and analyse the quality of the resulting system equilibria from the perspective of the global system performance, which can be quantitatively characterized by the price of anarchy (POA) [3–8]; (b) assuming the private information is not public, design and analyse mechanisms to guide individual competition results towards global optimization. Our study is belonging to the second case.

When the private information is not public, it is necessary to provide a mechanism that all customers commit their truthful information, because the facility owner requires the customers' private information for a global optimal decision. This research field is known as mechanism design, which began with the work of Leonid Hurwicz [9]. The mechanism design problem can be regarded as the process of solving the optimization problem of incomplete expression. In this case, the designer should first induce this expression and then solve the optimization problem [10]. There are two methods to induce the truthful information from agents: one class of mechanisms called direct revelation mechanisms (DRMs), which directly obtain truthful information by letting agents report their true types; the second called indirect mechanisms, the basic idea is to provide each agent with a choice of action and then assign a result to each action group; the strategy chosen by each agent will indirectly reflect their true type. Auctions are common examples of indirect mechanisms, and they provide several advantages over DRMs for the machine capacity allocation problems, such as the universal and anonymous auctions [11]. In the following subsection, we will specifically introduce the application of the auction mechanisms.

2.2. Auctions for Resource Allocation Problems. Auctions are market-based methods with an explicit set of rules determining prices and allocation of resource according to bids from the market participants [12]. The common auction formats include the Dutch auction, the English auction, the first price sealed bid auction, and the second price sealed bid auction (or Groves-Clarke-Vickery mechanism). These auction formats have been used to sell a wide range of objects, assets, and commodities [13]. With the development of auction theory, many new auctions formats have been designed for resource allocation problems in field of telecommunication system, traffic system, electric system, and so on. In telecommunication system, Cramton [14] presented a new combinatorial clock auction for governments to assign and price licenses for wireless communications. Cramton and Ockenfels [15] analysed and discussed the simultaneous ascending multiband auction for spectrum. In electric markets, Tang and Jain [16] designed auction mechanisms for the aggregators to procure stochastic resources. In traffic systems, Rassenti et al. [17] proposed a combinatorial auction mechanism for the allocation of airport time slots. Yang et al. [18] proposed an auction-based unified approach for prebooking urban logistics facilities.

A few papers also propose auction mechanisms for machine capacity allocation problems. For example, Kutanoglu and Wu [19] proposed a general auction mechanism for the job shop scheduling problem which uses the notion of multi-item combinatorial auction. They also concluded that there are strong links between Lagrangian-based decomposition and combinatorial auction. Wellman et al. [1] first developed a broad framework for using markets to solve decentralized scheduling problems. They presented an

ascending auction mechanism for capacity allocation problem in a single machine environment with multiple customers each having an order. They also proved that the equilibrium prices may not exist when using time slots as market goods. Dewan and Joshi [2] considered the distributed scheduling problem with the objective of minimizing earliness-tardiness penalties in a job shop environment. They presented an auction mechanism to schedule orders and proposed the iterative price adjustment method to reduce resource contention for the job. Hall and Liu [11] designed an auction mechanism to allocate the limited single machine production capacity among several competing customers, each having an order. Their market goods were time blocks and then proved that the equilibrium solution always exists but is not unique. Karabat and Yalcn [20] considered the pricing and production capacity allocation problem in the single machine environment and proposed an auction mechanism that uses the finished products as market goods for the private information case. They also developed alternative pricing strategies and conducted research towards the impact of the proposed auction mechanism on allocation efficiency. From the above, we see that all these studies present auction mechanisms for the capacity allocation problems in a single machine or job shop environment. In this paper, we aim to design an auction mechanism for the capacity allocation problem in two parallel machines with IPS restrictions.

2.3. Parallel Machine Scheduling with IPS Restrictions. There are many studies in the literature on parallel machine scheduling with IPS restrictions. Most of those focus on the development of the approximation algorithm (Polynomial Time Approximation Scheme, PTAS). For example, Kafura and Shen [21] considered a computing system model with several independent but identical processors, each with a limited amount of private memory. They proposed a largest memory time first (LMTF) algorithm to minimize the total completion time. Hwang et al. [22] considered the parallel machines scheduling problem in which the process service requests from various customers who are entitled to many different grades of service levels. They proposed a lowest grade and longest processing time (LG-LPT) algorithm for the problem. Ou et al. [23] developed a polynomial time approximation algorithm based on the LG-LPT algorithm. Li and Wang [24] extended the research problem of Kafura and Shen [21]; they studied the parallel machine scheduling problem with both release time and IPS constraints and developed a polynomial time approximation algorithm; Li et al. [25] considered parallel machine scheduling problems with batch processing and IPS restrictions and developed the polynomial time approximation algorithms for the same and different release times, respectively.

In this study, the winner determination problem has the following differences from the above researches. First, our winner determination problem features two concepts, the bid selection and two parallel machines' scheduling with IPS restrictions. Second, our winner determination problem aims to maximize the profit, whereas the above studies all adopt the traditional objective functions. In addition, the efficiency of the auction mechanism is very important, so we develop a fast heuristic to solve the winner determination problem in reasonable time.

3. Preliminaries

We formally describe the problem under study as follows: We have a set N of n competing customer orders interested in using the facility owner's production capacity. The facility owner possesses two parallel machines M_1 and M_2 which differ in their functionality but not in their processing speeds. We assume that M_2 can process all those orders that M_1 can process. The planning horizon spans a time period $t = 1, 2, \ldots, T$, and the facility owner sets per unit reserve values v_1 and v_2 for machines M_1 and M_2, respectively.

Associated with each customer order i is a processing time p_i, a deadline d_i, a revenue u_i, and a machine index $a_i = 1$ or 2. We assume that p_i, d_i, and u_i are integer valued. Let $N_1 = \{i | a_i = 1, i \in N\}$, and $N_2 = N/N_1$. Thus, M_1 can process those customer orders in N_1, and M_2 can process those customer orders in N.

In the decentralized decision-making environment, both the facility owner and the customers hold information privately. The facility owner's objective is to maximize its profit from selling the capacity to customers and holding any unallocated capacity at their reserve values; each customer's objective is to maximize its profit from getting its revenue through purchasing the facility owner production capacity to produce order.

4. Global Optimization Problem

In this section, we assume the information of the customer orders' processing requirements and revenues is public. First, we model the global optimization problem. Then we identify some properties of the optimal solution that can help us to develop heuristics for the winner determination problem.

The objective of the global optimization problem is to maximize the system profit. Without loss of generality, we assume that the customer orders are numbered in nondecreasing sequence of the deadlines, and the shortest processing times break the ties. Set three binary decision variables x_{it}^1, x_{it}^2, and y_{it}, where $x_{it}^1 = 1$ denotes that customer order $i \in N_1$ is accepted and completed at time t on machine M_1; otherwise $x_{it}^1 = 0$; $x_{it}^2 = 1$ denotes that customer order $i \in N_1$ is accepted and completed at time t on machine M_2; otherwise $x_{it}^2 = 0$; $y_{it} = 1$ denotes that customer order $i \in N_2$ is accepted and completed at time t on machine M_2; otherwise $y_{it} = 0$. We formulate the global optimization problem as the following integer linear programming model:

$$(GOP)\max \sum_{i\in N_1}\sum_{t=p_i}^{T}(u_i - v_1 p_i)x_{it}^1 + \sum_{i\in N_1}\sum_{t=p_i}^{T}(u_i - v_2 p_i)x_{it}^2 + \sum_{j\in N_2}\sum_{t=p_j}^{T}(u_j - v_2 p_j)y_{jt} + (v_1 + v_2)T \tag{1}$$

$$s.t. \sum_{t=p_i}^{T} x_{it}^1 + \sum_{t=p_i}^{T} x_{it}^2 \le 1, \quad \forall i \in N_1, \tag{2}$$

$$\sum_{t'=p_j}^{T} y_{jt'} \le 1, \quad \forall j \in N_2, \tag{3}$$

$$\sum_{i\in N_1}\sum_{s=\max\{p_i,t\}}^{\min\{t+p_i-1,T\}} x_{it}^2 + \sum_{j\in N_2}\sum_{s'=\max\{p_j,t\}}^{\min\{t+p_j-1,T\}} y_{js'} \le 1, \quad 1 \le t \le T, \tag{4}$$

$$\sum_{i\in N_1}\sum_{s=\max\{p_i,t\}}^{\min\{t+p_i-1,T\}} x_{it}^1 \le 1, \quad 1 \le t \le T, \tag{5}$$

$$\sum_{t=1}^{T} t x_{it}^1 \le d_i, \quad \forall i \in N_1, \tag{6}$$

$$\sum_{t=1}^{T} t x_{it}^2 \le d_i, \quad \forall i \in N_1, \tag{7}$$

$$\sum_{t=1}^{T} t y_{jt} \le d_j, \quad \forall j \in N_2. \tag{8}$$

In the above model GOP, Constraints (2) and (3) state that the accepted customer orders should be processed exactly once. Constraints (4) and (5) state that, in each time slot, at most one customer order can be processed on machines M_1 and M_2, respectively. Constraints (6), (7), and (8) make sure that the accepted customer orders are completed before their deadlines.

Now, we present some optimal properties of the solution for the global optimization problem.

Lemma 1. *For the global optimization problem, there is an optimal schedule in which the accepted customer orders are processed in the earliest due date (EDD) order on both machines.*

Proof. Assuming that π^* is an optimal solution for the global optimization problem, in π^*, there are two adjacent customer orders i and j, and j is processed after i, and $d_i > d_j$. Let t be the start time of i. Perform a neighbor-pair exchange on order i and order j, and get a new solution $\pi\prime$.

In π^*, the completion times of i and j are $t + p_i \le d_i$ and $t + p_i + p_j \le d_j$, respectively. In $\pi\prime$, the completion times of i and j are $t + p_i + p_j$ and $t + p_j$, respectively. As we know, $d_i > d_j$, so $t + p_i + p_j \le d_j < d_i$, $t + p_j < d_j$. Customer orders i and j can also be completed before their deadlines; the new solution π' is also an optimal solution. Therefore, we reach the conclusion. □

Lemma 1. *It tells us that no matter which set the accepted customer orders come from, they should be scheduled according to the EDD rule on each machine.*

Lemma 2. *For the global optimization problem, there is an optimal schedule in which if at least one order of S_i is processed on one of the machines, then customer order i is also accepted for processing, where $S_i = \{k | p_k > p_i, d_k \le d_i, u_k \le u_i \text{ and } a_k \ge a_i\}$, $i \in N$.*

Proof. Assume that π^* is an optimal solution for the global optimization problem, and customer order i is not in π^*, but customer order k in the set S_i is accepted and completed at time t in π^*, where $S_i = \{k | p_k > p_i, d_k \le d_i, u_k \le u_i \text{and} a_k \ge a_i\}$, $i \in N$ and $t \le d_k$. Knowing that $a_k \ge a_i$, the machine that can process k can also process i, so we can replace k by i. Denote the new solution as π'.

In the new solution $\pi\prime$, the completion time of i is t, we see that $d_k \le d_i$, so $t \le d_i$, and customer order i can also be processed by its deadline in $\pi\prime$. As known $p_k > p_i$ and $u_k \le u_i$, so the profit of $\pi\prime$ is larger than that of π^*, a contradiction arises. Therefore, we reach the conclusion. □

Lemma 2. *It is implied that if a solution only includes customer order k but not i, then it is not optimal.*

5. Private Information Problem

In this section, we consider the capacity allocation problem, in which the customers keep all the information private, except for the processing times and deadlines. We propose an iterative ascending auction mechanism to solve the private information case. In the following, we will elaborate our auction mechanism from auction protocol, pricing problem, and winner determination problem.

5.1. Auction Protocol. In our iterative ascending auction mechanism, the auctioneer is the facility owner who owns two parallel machines that have the same processing speed but differ in their functionality. The bidders are the customers; each of them has an order to process. The market goods are time blocks; each customer can bid for one single time block at once.

In the first round, the facility owner sets the initial ask price α_i for customers, and (a) each customer sends a bid for processing order so as to maximize his own profit; the bids of the customers are defined as $B_i = [p_i, d_i, a_i, \alpha_i]$, where p_i and d_i are the order's processing time and deadline. Once the processing time and deadlines of the customer orders are confirmed in the first round, it cannot be changed. (b) Then the facility owner collects all bids and determines which bids to accept in order to maximize his own profit and updates the ask price α_i for each customer.

Step (a) and (b) will be iterated; as the ask prices continue to rise gradually, customers gradually withdraw from competition. Once any customer withdraws from the competition, he shall not return to participate in the auction again. If none of the customers submits a new bid, then the auction is terminated.

The design of an optimal auction mechanism (i.e., pricing policy and winner determination algorithm) should fulfil the following properties:

(1) Individual rationality means that if a customer participates in the auction mechanism, his profit is at least as high as the profit that he is not participating.

(2) Incentive compatibility means that bidding by the truthful order information is a dominant strategy for each customer.

(3) Global optimization means the winner determination problem should be a global optimization problem when all the customers bid by their true values.

(4) Computational efficiency means the auction should reach closure in reasonable time and at reasonable computational expense.

Nisan et al. [26] analyse that the second price sealed auction satisfies the above properties (1) and (2) in combinatorial auctions. However, a sealed auction is less efficient and profitable than an ascending auction in production scheduling problems (Hall and Liu 2015). What is more, in the private value model, an ascending auction is equivalent to a second price sealed auction. That is because, when the value of market goods for each customer is independent of the others, each customer's dominant strategy is to keep in bidding until the ask price reaches his value, regardless of whether he can observe the process of the auction. At last, the winner will be the one who has the highest value. So the ascending auction mechanism can also be designed to satisfy properties (1) and (2) in our production capacity allocation problem.

In our problem, the winner determination problem is NP hard, so the above properties (3) and (4) are incompatible. When the problem size is small, the optimal solution can be obtained in a limited time. However, when the problem size is large, the optimal solution cannot be obtained in a reasonable time. Therefore, we use approximate optimal solution in exchange for high computational efficiency.

For our auction mechanism, the purpose of the pricing problem is to encourage all the customers to reveal truthful processing requirements, such as machine indexes, processing times, and deadlines, and make the customers' profits nonnegative; the winner determination problem is NP hard; we aim to develop a fast heuristic to ensure that the auction reaches closure in reasonable time and at reasonable computational expense.

5.2. Pricing Problem. We define the ask price in round $k-1$ for each customer i ($i \in N$) as α_i^k. It consists of two parts: (a) current price β_i^{k-1}, which reflects the current capacity scarcity under the generated schedule in round $k-1$, and (b) price increment γ_i^k , which reflects the expected impact on capacity scarcity caused by potential bids.

First, we define the current price β_i^k in round k for customer i, based on the generated schedule in round $k-1$:

(1) If no time slots before d_i are allocated in round $k-1$ on both machines, the $\beta_i^k = \nu_1 + \nu_2/2$.

(2) If the time slots before d_i are allocated to orders $\{1, 2, \ldots, n_i\}$ with prices $\alpha_1^{k-1}, \alpha_2^{k-1}, \ldots, \alpha_{n_i}^{k-1}$ in round $k-1$ on both machines, then β_i^k is equal to the weighted average of the bid prices of all allocated time slots and the total reserved value of all unallocated time slots. It is clear that the last orders on both machines may be partially processed in $[0, d_i]$. To simplify the expression of β_i^k, let $\overline{p}_j^{k-1}$ be the length of the time slots within $[0, d_i]$ that were allocated to order j on machine a_i in round $k-1$. If order j is the last order on one of the machines before d_i, then, $\overline{p}_j^{k-1} = p_j - \max\{0, C_j^{k-1} - d_j\}$; else $\overline{p}_j^{k-1} = p_j$, where C_j^{k-1} is the completion time of the order j. Let I_i^{k-1} be the idle times before d_i on the two machines at round $k-1$. Then β_i^k is equal to a weighted average price:

$$\beta_i^k = \frac{2\sum_{h=1}^{n_i} \alpha_h^{k-1} \overline{p}_h^{k-1} + I_i^{k-1}(\nu_1 + \nu_2)}{2d_i}. \quad (9)$$

Then, we define the price increment γ_i^k in round k for customer i; it is based on (a) the flexibility of order processing related to its deadline and machine index, if the flexibility of order processing impact on resource scarcity is greater and, consequently, the price increment increases; (b) the number of bids received by this time block that contains each individual time slot. The definition is as follows:

(1) If no time slots before d_i are allocated in round $k-1$, then

$$\gamma_i^k = \rho \frac{p_i a_i}{d_i}. \tag{10}$$

(2) If the time slots before d_i are allocated in round $k-1$, then

$$\gamma_i^k = \rho \left(\frac{p_i a_i}{d_i} + \frac{\sum_{t=1}^{d_i} D_t^{k-1}}{N d_i} \right). \tag{11}$$

ρ are predetermined constants, called price adjustment factors; a_i, p_i, and d_i are order $i\prime$ s machine index, processing time, and deadline; D_t^{k-1} is the number of bids that contain time slot t in round $k-1$.

The purpose of the pricing problem is to encourage the customers to reveal truthful machine indexes, processing times, and deadlines. We assume that the customer would not pretend that he has a smaller machine index and processing time and a greater deadline. See, for example, a customer j having an order with a machine index $a_j = 2$, a processing time $p_j = 5$, and a deadline $d_j = 10$. It is obvious that he would not pretend $p_j < 5$, or $d_j > 10$. If he pretends $a_j = 1$, his order may be assigned to machine M_1 or M_2; however, machine M_1 cannot process a high index order, so it is impossible for him to bid by a false smaller machine index. So, in our pricing problem, we only need to ensure that the customer would not bid by a false larger machine index and processing time and a shorter deadline. From the above definition of price, we can see that the initial ask price for customer i $(i \in N)$ is $\alpha_i = (v_1 + v_2)/2 + \rho p_i a_i / d_i$; it is positively correlated with machine index and processing time and negatively correlated with deadline, so each customer will not bid by a larger machine index and processing time or a shorter deadline.

5.3. Winner Determination Problem. In this section, we formulate our winner determination problem as the facility owner's profit maximization problem by selecting the customer bids and scheduling the accepted bids to the two machines synchronously. This subsection is structured as follows: first, we formulate the winner determination problem, and the Lagrangian relaxation method is used to determine the subset of the accepted bids; then we use a heuristic to construct a feasible schedule with these accepted bids.

5.3.1. Lagrangian Relaxation of the Winner Determination Problem. The integer linear programming model of the winner determination problem is described as follows:

$$\begin{aligned}(WDP)\max \quad & \sum_{i\in N_1}\sum_{t=p_i}^{T}(\alpha_i - v_1 p_i)x_{it}^1 + \sum_{i\in N_1}\sum_{t=p_i}^{T}(\alpha_i - v_2 p_i)x_{it}^2 \\ & + \sum_{j\in N_2}\sum_{t=p_j}^{T}\left(\alpha_j - v_2 p_j\right)y_{jt} + (v_1+v_2)T \\ & s.t. \quad (1),\ (2),\ (3),\ (4),\ (5),\ (6),\ (7).\end{aligned} \tag{12}$$

Relax the constraints (3) and (4), set $\lambda = \begin{pmatrix} \lambda_{11} & \lambda_{21} \\ \lambda_{12} & \lambda_{22} \\ \cdots & \cdots \\ \lambda_{1T} & \lambda_{2T} \end{pmatrix}$ to be a vector of corresponding nonpositive multipliers, and get the following Lagrangian problem (LR):

$$\begin{aligned}(LR)\, L(\lambda) = \max & \sum_{i\in N_1}\sum_{t=p_i}^{T}(\alpha_i - v_1)p_i x_{it} + \sum_{i\in N_1}\sum_{t=p_i}^{T}(\alpha_i - v_2 p_i)y_{it} \\ & + \sum_{j\in N_2}\sum_{t'=p_j}^{T}\left(\alpha_j - v_2 p_j\right)z_{jt} + (v_1+v_2)T \\ & + \sum_{t=1}^{T}\lambda_{1t}\left(\sum_{i\in N_1}\sum_{s=\max\{p_i,t\}}^{\min\{t+p_i-1,T\}} x_{is} + \sum_{j\in N_2}\sum_{s'=\max\{p_j,t\}}^{\min\{t+p_j-1,T\}} z_{js} - 1\right) \\ & + \sum_{t=1}^{T}\lambda_{2t}\left(\sum_{i\in N_1}\sum_{s=\max\{p_i,t\}}^{\min\{t+p_i-1,T\}} y_{is} - 1\right) \\ s.t \quad & (1),\ (2),\ (5),\ (6) \text{ and } (7).\end{aligned} \tag{13}$$

LR can be rewritten as

$$
\begin{aligned}
(LR)\, L(\lambda) = \max & \sum_{i\in N_1}\sum_{t=p_i}^{T}\left(\alpha_i - v_1 p_i + \sum_{s=\max\{1,t-p_i+1\}}^{t}\lambda_{1s}\right)x_{it} \\
& + \sum_{i\in N_1}\sum_{t=p_i}^{T}\left(\alpha_i - v_2 p_i + \sum_{s=\max\{1,t-p_i+1\}}^{t}\lambda_{2s}\right)y_{it} \\
& + \sum_{j\in N_2}\sum_{t=p_j}^{T}\left(\alpha_j - v_2 p_j + \sum_{s=\max\{1,t-p_j+1\}}^{t}\lambda_{1s}\right)z_{jt} \\
& + (v_1+v_2)T - \sum_{t=1}^{T}\lambda_{1t} - \sum_{t=1}^{T}\lambda_{2t} \\
s.t \quad & (1),\ (2),\ (5),\ (6) \text{ and } (7).
\end{aligned}
\tag{14}
$$

Note that $L(\lambda)$ can be decomposed into

$$
\begin{aligned}
L(\lambda) = & \sum_{i\in N_1} L_i^1(\lambda) + \sum_{i\in N_2} L_i^2(\lambda) \\
& + \left((v_1+v_2)T - \sum_{t=1}^{T}\lambda_{1t} - \sum_{t=1}^{T}\lambda_{2t}\right).
\end{aligned}
\tag{15}
$$

Here,

$$
\begin{aligned}
L_i^1(\lambda) = \max & \sum_{t=p_i}^{T}\left(\begin{array}{l}\left(\alpha_i - v_1 p_i + \sum_{s=\max\{1,t-p_i+1\}}^{t}\lambda_{1s}\right)x_{it} \\ +\left(\alpha_i - v_2 p_i + \sum_{s=\max\{1,t-p_i+1\}}^{t}\lambda_{2s}\right)y_{it}\end{array}\right), \\
s.t & \sum_{t=p_i}^{T} x_{it} + \sum_{t=p_i}^{T} y_{it} \le 1, \sum_{t=1}^{T} t x_{it} \le d_i, \sum_{t=1}^{T} t y_{it} \le d_i, \\
L_i^2(\lambda) = \max & \sum_{t'=p_i}^{T}\left(\alpha_i - v_2 p_i + \sum_{s=\max\{1,t-p_i+1\}}^{t}\lambda_{1s}\right)z_{it'}, \\
s.t & \sum_{t'=p_i}^{T} z_{it'} \le 1, \sum_{t=1}^{T} t z_{it} \le d_i.
\end{aligned}
\tag{16}
$$

$L_i^1(\lambda)$ and $L_i^2(\lambda)$ are two subproblems of bid i; it is easy to see that

$$
\begin{aligned}
L_i^1(\lambda) &= \max\left\{0, \max_{p_i\le t\le d_i}\left(\begin{array}{l}\left(\alpha_i - v_1 p_i + \sum_{s=\max\{1,t-p_i+1\}}^{t}\lambda_{1s}\right), \\ \left(\alpha_i - v_2 p_i + \sum_{s=\max\{1,t-p_i+1\}}^{t}\lambda_{2s}\right)\end{array}\right)\right\}, \\
L_i^2(\lambda) &= \max\left\{0, \max_{p_i\le t\le d_i}\left(\alpha_i - v_2 p_i + \sum_{s=\max\{1,t-p_i+1\}}^{t}\lambda_{1s}\right)\right\}.
\end{aligned}
\tag{17}
$$

The Lagrangian dual of (LR) is defined as follows:

$$
(D)\, L_D = \min_{\lambda\le 0} L(\lambda). \tag{18}
$$

Then, the subgradient method is used to determine λ so as to find optimal or near-optimal solutions for LD. The subgradient method is presented as follows.

(1) Subgradient Algorithm.

Step 1: set K and σ. Let $k = 0$; set λ^0 and μ^0.

Step 2: calculate the original problem's feasible solution, and record its objective value as $\overline{Z}$.

Step 3: calculate the Lagrangian problem $L^k(\lambda^k)$, and record its solution x^k, y^k, and z^k.

Step 4: let $\lambda_1^{k+1} = \lambda_1^k + \beta_1^k G_1(x^k, y^k, z^k)/\|G_1(x^k, y^k, z^k)\|$, where $G_1(x^k, y^k, z^k)$ is the subgradient at point (x^k, y^k, z^k) and

$$
\begin{aligned}
G_1\left(x^k, y^k, z^k\right) = & \sum_{i\in N_1}\sum_{s=\max\{p_i,t\}}^{\min\{t+p_i-1,T\}} x_{is} \\
& + \sum_{j\in N_2}\sum_{s_1=\max\{p_j,t\}}^{\min\{t+p_j-1,T\}} z_{js'} - 1.
\end{aligned}
\tag{19}
$$

Set $\mu_1^k = \sigma\mu_1^{k-1}$, and β_1^k is the step size, $\beta_1^k = \mu_1^k(\overline{Z} - L^k(u^k))/\|G_1(x^k, y^k, z^k)\|$.

Let $\lambda_2^{k+1} = \lambda_2^k + \beta_2^k G_2(x^k, y^k, z^k)/\|G_2(x^k, y^k, z^k)\|$, where $G_2(x^k, y^k, z^k)$ is the subgradient at point (x^k, y^k, z^k) and

$$
G_2\left(x^k, y^k, z^k\right) = \sum_{i\in N_1}\sum_{s=\max\{p_i,t\}}^{\min\{t+p_i-1,T\}} y_{is} - 1. \tag{20}
$$

Set $\mu_2^k = \sigma\mu_2^{k-1}$ and $\beta_2^k = \mu_2^k(\overline{Z} - L^k(u^k))/\|G_2(x^k, y^k, z^k)\|$.

Step 5: if $k < K$, let $k \leftarrow k+1$; go to Step 3; otherwise, stop.

Li and Sun [27] demonstrated that, when using the above expression to calculate the step size, the subgradient algorithm will be convergent and it can converge to an optimal solution for the dual problem with suitable parameters. In the subgradient algorithm, the parameters need to be predetermined experimentally. The iterations K should ensure sufficient convergence of the algorithm; the step adjustment factors μ_1^k and μ_2^k character the relationship between the quality of solutions and the step size of iterations; if the values are too large, convergence oscillation will occur; if the values are too small, convergence speed will decrease. Set the initial parameter $\mu^0 = (1,1)$, $\lambda^0 = \begin{pmatrix} -1,-1,\ldots,-1 \\ -1,-1,\ldots,-1 \end{pmatrix}^T$. By the algorithm, we can obtain an upper bound $L^K(\lambda^K)$ for the winner determination problem, while vectors x^K, y^K, and z^K indicate which bids are accepted for processing. Using the upper bound $L^K(\lambda^K)$, we can evaluate the performance of the following heuristic.

5.3.2. Heuristic of the Winner Determination Problem. From the subgradient algorithm, we get a preliminary result of bids acceptance; however, the result may not be a festival solution for the winner determination problem, so we design a heuristic to construct a feasible schedule. The heuristic is mainly based on the Lagrangian relaxation technique, and it is described as follows.

(1) Heuristic.

Step 1: let $N\prime$ denote the subset of accepted bids in N by the subgradient algorithm; let $N\prime\prime = N/N\prime$. Index the bids of $N\prime$ in nondecreasing order of their due dates and break ties according to the shortest processing time.

Step 2: for bid $j = 1, 2, \ldots, |N\prime|$, perform Steps 2-1 to 2-2.

2-1: if $a_j = 1$, perform Steps 2-1-1 to 2-1-2.
2-1-1: if bid j can be completed before its deadline on M_1, then assign it to M_1, next j.
2-1-2: if bid j cannot be completed before its deadline on M_1, then discard bid j and let $N\prime\prime = N\prime\prime \cup \{j\}$, next j.
2-2: if $a_j = 2$, perform Steps 2-2-1 to 2-2-3.
2-2-1: if bid j can be completed before its deadline on M_1 and M_2, then assign it to one machine on which bid j is finished as close to its due date as possible, next j.
2-2-2: if bid j can be completed before its deadline on M_1 or M_2, then assign it to the machine on which bid j is finished on time, next j.
2-2-3: if bid j cannot be completed before its deadline on M_1 or M_2, then discard bid j and let $N\prime\prime = N\prime\prime \cup \{j\}$, next j.

Step 3: select successively one bid l with $b_l/p_l = \max\{b_k/p_k | k \in N''\}$. Calculate $S_l = \{k | p_k > p_l, d_k \le d_l, u_k \le u_l \text{ and } a_k \ge a_l, k \in N'\}$; if $S_l \neq \varnothing$, then perform Steps 3-1 to 3-2; else let $N'' = N''/l$.

3-1: insert bid l in all the positions on M_1 and record their objective values. Let $N'' = N''/\{l\}$ and if the maximum objective value among the generated schedules is larger than the current schedule, then take the generated schedule as the current schedule.
3-2: insert bid l in all the positions on M_1 or M_2; refer to Step 3-1.

Step 4: if $N'' \neq \varnothing$, go to Step 3; else, stop the heuristic.

In the heuristic of the winner determination problem, we construct a feasible solution in Step 2. In order to make the solution converge to an optimal solution for the winner determination problem, we continuously improve the solution by applying greedily some searching techniques: First, we use Lemma 2 to check whether the bids are likely to be inserted into the machines. Then we add the unaccepted bids to the parallel machines greedily. The computing cost of the winner determination problem is dominated by the solution of the Lagrangian dual problem, so the running time of this problem is $O(nT)$.

6. Computational Experiments

In this section, we conducted computational experiments to analyse the computational efficiency and effectiveness of the ascending auction mechanism. All experiments were run on an Intel 2.9 GHz octa-core processor with 32G RAM. In the following, we present the experimental schemes and the discussion of the results.

6.1. Data Generation. To test the performance of the auction mechanism extensively, we randomly generate instances of the problem by varying the problem size and parameters that may affect the analysis of the auction mechanism. First, we select $n \in \{25, 50, 100\}$ and randomly generate the processing time p_i $(i \in N)$ from the integer uniform distribution in the interval $[1, 100]$. Let the total processing time $P = \sum_{i \in N} p_i$, for each instance with P, the facility owner's capacity $T = \lfloor \eta P/2 \rfloor$, where $\eta \in \{0.5, 0.7, 0.9\}$, and the reserve values v_1 and v_2 are randomly generated from the integer uniform distribution in the interval $[1, 10]$, respectively. For each customer order i, we randomly generate its deadline d_i from the integer uniform distribution in the interval $[\max\{p_i, \omega T\}, T]$, where $\omega \in [0.2, 0.5, 0.8]$, revenue u_i from the integer uniform distribution in the interval $[p_i \max\{v_1, v_2\}, 1000]$, and machine index a_i from $\{1, 2\}$. Then, for the pricing problem in Subsection 5.2, we define the price adjustment factors $\rho \in \{0.5, 1.0, 2.0\}$. For each $3 \times 3 \times 3 \times 3 = 81$ possible combinations of parameters, we randomly generate 10 problem instances, for a total of 810.

6.2. Analysis of Heuristic of the Winner Determination Problem. To ensure the auction reach closure in reasonable time and at reasonable computational expense, we use a

Table 1: The performance of heuristic of the winner determination problem.

η	ω	R (%)		
		$n = 25$	$n = 50$	$n = 100$
0.5	0.2	97.48	94.46	90.12
	0.5	95.58	92.29	89.53
	0.8	96.53	91.73	87.38
0.7	0.2	95.34	89.67	87.69
	0.5	96.53	92.48	89.73
	0.8	94.82	90.53	90.45
0.9	0.2	93.91	88.42	90.10
	0.5	96.12	93.58	89.24
	0.8	97.73	91.67	87.30
Average		96.00	91.65	89.06

Table 2: The performance of the auction mechanism with $n = 25, 50$.

η	ω	ρ	POA (%)			Round		
			$n = 25$	$n = 50$	$n = 100$	$n = 25$	$n = 50$	$n = 100$
0.5	0.2	0.5	94.09	89.40	85.13	15.0	24.5	45.7
		1.0	93.03	87.14	85.92	12.0	20.2	28.3
		2.0	93.02	85.18	83.59	10.3	19.6	24.9
	0.5	0.5	91.68	87.60	83.63	14.0	21.6	39.9
		1.0	92.18	88.08	84.51	12.0	19.3	26.8
		2.0	90.78	86.28	81.29	11.3	19.4	21.9
	0.8	0.5	85.46	85.10	84.33	17.6	30.7	41.3
		1.0	92.49	84.35	83.58	18.3	26.4	32.6
		2.0	93.22	82.70	80.49	10.0	19.8	28.5
0.7	0.2	0.5	92.94	89.36	82.60	15.7	20.5	35.9
		1.0	94.86	90.05	83.78	13.3	20.2	27.5
		2.0	93.85	87.11	82.79	11.7	19.0	25.4
	0.5	0.5	90.58	86.19	83.55	18.8	30.1	39.3
		1.0	93.62	87.21	83.69	16.7	26.4	30.5
		2.0	93.24	84.62	82.22	14.6	24.0	31.6
	0.8	0.5	93.45	83.22	79.88	23.0	31.1	44.6
		1.0	93.34	85.58	81.36	21.7	31.8	39.5
		2.0	96.35	88.31	84.91	15.3	24.4	30.6
0.9	0.2	0.5	94.76	91.14	86.28	18.7	33.9	42.7
		1.0	96.52	86.47	85.81	15.0	24.8	37.8
		2.0	98.79	85.19	83.36	14.7	25.0	29.1
	0.5	0.5	96.61	87.06	85.67	18.3	30.8	38.4
		1.0	97.67	83.59	82.02	15.7	23.4	32.3
		2.0	98.47	89.04	80.07	16.7	25.4	28.2
	0.8	0.5	92.96	83.17	82.55	34.7	40.3	48.5
		1.0	93.26	82.32	81.84	28.0	35.6	39.9
		2.0	95.89	85.08	81.67	22.7	31.8	37.5
Average			93.82	86.24	83.20	16.9	25.9	34.4

heuristic solving the winner determination problem in exchange for high computational efficiency. In this subsection, we evaluate the performance of this heuristic. First, we use the Lagrangian relaxation technique (Subsection 5.3.1) to establish an upper bound on the facility owner's maximum profit, which is denoted as UB_f. Let OPT_f and H_f denote the optimal solution value and the heuristic solution, respectively. For the small-sized instances with $n = 25$, we compare the heuristic solution value to the optimal solution value, where $R = H_f/OPT_f$; for the large-sized instances with $n = 50, 100$, we compare the heuristic solution value to the upper bound, where $R = H_f/UB_f$.

The computational results are summarized in Table 1. First, the overall mean values of R for the cases $n = 25, 50$, and 100 are 96.00%, 91.65%, and 89.06%, respectively. The heuristic of the winner determination problem performs well across different problem sizes and parameters. Second, we notice that the mean value of R for the large-sized problem is significantly smaller than that of the small-sized problem. This is because the value of R for the large-sized problems is underestimated due to the gaps between the optimal solutions and the upper bounds. Third, when the value of the facility owner's capacity parameter η is bigger, different values of the customer order's deadline parameter

ω will have a more significant impact on the performance of the heuristic. This may indicate that the performance of the heuristic is mainly determined by the scarcity of resource. From the above, we can conclude that the heuristic of the winner determination problem is effective in finding good solutions rapidly.

6.3. Analysis of the Auction Mechanism. To evaluate the performance of the auction mechanism, we define the POA in our study as the percentage deviation of the system value of the auction outcome from that of the global system optimum. For the large-sized instances, it is impossible to get the global optimal solution in a reasonable time with CPLEX, so we use an upper bound instead of the global optimal solution.

The computational results with different parameters are summarized in Table 2. First, the mean values of POA for the case where $n = 25, 50$, and 100 are 93.82%, 86.24%, and 83.2%, respectively. As the number of customers decreases, the auction performs better. Second, as η increases from 0.5 to 0.9, ω decreases from 0.8 to 0.2, and the auction mechanism performs better. This shows that the fewer the conflicts in resource requirements are, the better the auction performs. Third, the mean number of rounds for the auction to reach closure is 25.7. When the number of customers is larger, and the price adjustment factor ρ is smaller, the auction needs more rounds to reach closure. However, the auction does not perform better when the number of rounds is increased. This may result from the combinatorial optimization characteristics of the machine capacity's allocation problem.

7. Conclusion

In this study, we present an iterative ascending auction mechanism for the scare production capacity allocation problem in two parallel machines with IPS restrictions. In our auction, the customers do not have to make decisions with all the information; they only need to decide whether to bid or not. The ask prices are updated adaptively by the facility owner which can make the customers bid by truthful processing requirements and keep on bidding until the ask price reaches their real revenues. The winner determination problem integrates the capacity allocation and scheduling decisions in two parallel machines with IPS restrictions. The proposed heuristic of winner determination problem is efficient in finding good solutions. This ensures our auction reaches closure in a reasonable time. Overall, our auction mechanism is an effective method to allocation the capacity in two parallel machines with IPS restrictions.

Data Availability

The data supporting the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported in part by the National Natural Science Foundation of China under Grant 71871118.

References

[1] M. P. Wellman, W. E. Walsh, P. R. Wurman, and J. K. MacKie-Mason, "Auction protocols for decentralized scheduling," *Games and Economic Behavior*, vol. 35, no. 1-2, pp. 271–303, 2001.

[2] P. Dewan and S. Joshi, "Auction-based distributed scheduling in a dynamic job shop environment," *International Journal of Production Research*, vol. 40, no. 5, pp. 1173–1191, 2002.

[3] E. Koutsoupias and C. Papadimitriou, C. Meinel, "Worst-case equilibria, stacs 99," in *STACS 1999, 16th Annual Symposium on Theoretical Aspects of Computer Science*, S. Tison, Ed., vol. 1563, Berlin, Germany, Springer, 1999.

[4] B. Heydenreich, R. Mu¨ller, and M. Uetz, "Games and mechanism design in machine scheduling--an introduction," *Production and Operations Management*, vol. 16, no. 4, pp. 437–454, 2007.

[5] G. Christodoulou, E. Koutsoupias, and A. Nanavati, "Coordination mechanisms," in *Automata, Languages and Programming*vol. 3142, Berlin, Germany, Springer, 2004.

[6] N. Immorlica, L. Li, V. S. Mirrokni, and S. S. Andreas, "Coordination mechanisms for selfish scheduling," *Theoretical Computer Science*, vol. 410, pp. 1589–1598, 2009.

[7] K. Lee, J. Y.-T. Leung, and L. M. Pinedo, "Coordination mechanisms for parallel machine scheduling," *European Journal of Operational Research*, vol. 220, pp. 305–313, 2012.

[8] Q. Chen, L. Lin, Z. Tan, and Y. Yan, "Coordination mechanisms for scheduling games with proportional deterioration," *European Journal of Operational Research*, vol. 263, pp. 380–389, 2017.

[9] L. Hurwicz, "Optimality and information efficiency in resource allocation processes," in *Mathematical Methods in the Social Science*, K. J. Arrow and S. Karlin, Eds., Stanford University Press, California, CL, USA, 1960.

[10] D. Condorelli, "Market and non-market mechanisms for the optimal allocation of scarce resources," *Games and Economic Behavior*, vol. 82, pp. 582–591, 2013.

[11] G. N. Hall and Z. X. Liu, "Market good flexibility in capacity auction," *Production and Operations Management*, vol. 22, no. 2, pp. 459–472, 2013.

[12] R. P. McAfee and J. McMillan, "Auctions and bidding," *Journal of Economic Literature*, vol. 25, no. 2, pp. 699–738, 1987.

[13] V. Krishna, *Auction Theory*, Academic Press, Amsterdam, Netherlands, 2nd edn. edition, 2010.

[14] P. Cramton, "Spectrum auction design," *Review of Industrial Organization*, vol. 42, no. 2, pp. 161–190, 2013.

[15] P. Cramton and A. Ockenfels, "The German 4G spectrum auction: design and behaviour," *Economic Journal*, vol. 127, no. 605, pp. 305–324, 2017.

[16] W. Tang and R. Jain, "Stochastic resource auctions for renewable energy integration," in *Proceedings of the 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 345–352, Monticello, IL, USA, September 2011.

[17] S. J. Rassenti, V. L. Smith, and R. L. Bulfin, "A combinatorial auction mechanism for airport time slot allocation," *The Bell Journal of Economics*, vol. 13, no. 2, pp. 402–417, 1982.

[18] K. Yang, R.-R. Mireia, and M. Menndez, "An auction-based approach for prebooked urban logistics facilities," *Omega, The International Journal of Management Science*, vol. 89, pp. 193–211, 2019.

[19] E. Kutanoglu and S. D. Wu, "On combinatorial auction and lagrangean relaxation for distributed resource scheduling," *IIE Transactions*, vol. 31, no. 9, pp. 813–826, 1999.

[20] S. Karabat and Z. B. Yalcn, "An auction mechanism for pricing and capacity allocation with multiple products," *Production and Operations Management*, vol. 23, no. 1, pp. 81–94, 2014.

[21] D. G. Kafura and V. Y. Shen, "Task scheduling on a multiprocessor system with independent memories," *SIAM Journal on Computing*, vol. 6, pp. 167–187, 1977.

[22] H. C. Hwang, S. Y. Chang, and K. Lee, "Parallel machine scheduling under a grade of service provision," *Computers & Operations Research*, vol. 31, no. 12, pp. 2055–2061, 2004.

[23] J. Ou, J. Y. T. Leung, and C. L. Li, "Scheduling parallel machines with inclusive processing set restrictions," *Naval Research Logistics*, vol. 55, no. 4, pp. 328–338, 2008.

[24] C. L. Li and X. L. Wang, "Scheduling parallel machines with inclusive processing set restrictions and job release times," *European Journal of Operational Research*, vol. 200, pp. 702–710, 2010.

[25] S. G. Li, "Parallel batch scheduling with inclusive processing set restrictions and non-identical capacities to minimize makespan," *European Journal of Operational Research*, vol. 260, pp. 12–20, 2017.

[26] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, *Algorithmic Game Theory*, Cambridge University Press, New York, NY, USA, 2007.

[27] D. Li and X. L. Sun, *Nonlinear Integer Programming*, Springer, New York, NY, USA, 2006.

Hindawi
Mathematical Problems in Engineering
Volume 2021, Article ID 8324926, 11 pages
https://doi.org/10.1155/2021/8324926

Research Article

Control Strategy of Microgrid Inverter Based on H_∞ State Feedback Repeated Deadbeat Control

Ren Xie, Yougui Guo, and Yonghong Lan

College of Automation and Electronic Information, Xiangtan University, Xiangtan Hunan, China

Correspondence should be addressed to Yougui Guo; 719983182@qq.com

Received 19 September 2021; Accepted 18 November 2021; Published 21 December 2021

Academic Editor: Guoqiang Wang

Aiming at the voltage distortion at the microgrid public connection point caused by nonlinear loads, a H_∞ state feedback deadbeat repetitive control strategy is proposed to rectify the total harmonic distortion of the output voltage. Firstly, through establishing the state space of the repetitive controller, introducing state feedback, combining the H_∞ control theory, and reformulating the system stability problem as a convex optimization problem with a set of linear matrix inequality (LMI) constraints to be solved, high stability control accuracy can be guaranteed and antiharmonic interference strengthened. Secondly, by introducing deadbeat control technology to improve the transient response speed of the system, changes in output voltage caused by load changes can be quickly compensated. Compared with the existing methods, the designed control method has the advantages of good stability, low harmonic content, and fast convergence speed, and the results are easier to verify. Finally, the simulation verifies the effectiveness of the proposed control strategy.

1. Introduction

In recent years, the microgrid has received wide attention for its unique form of maximizing the flexibility and advantages of distributed generation systems (DG) [1–3]. Its internal energy conversion is mainly conducted via power electronics, and the inverter is the core link of the distributed generation system, its operating state is related to the performance of the whole system, and z-source inverter is widely used because of its function of voltage-up and step-down conversion. When the microgrid runs in the island mode, affected by the harmonic current generated by the nonlinear loads, the voltage at the Point of Common Coupling is thus distorted, leading to the degradation of the supply voltage quality and affecting the normal operation of the load and inverter [4–6]. Therefore, it is of great practical significance to study a control strategy for reducing the total harmonic distortion (THD) of the output voltage in a state of satisfying the demand for nonlinear loads.

Proportional integral (PI) control, though deficient [7, 8] for harmonic suppression, is widely used in microgrid inverters due to its easily realized structure, and the repetitive control, taking the advantage of its convenient implementation, easily realized structure, and high efficient waveform control, is widely applied to control the inverters, but the response speed is slow [9]. To enhance the robustness of the system, [10] proposes a combined controller for repetitive and sliding mode control and adds feedforward control to improve its dynamic properties, but the design of phase advance units and low-pass filters is complex. Document [11] proposed a method based on mixed sensitivity to determine repetitive control parameters. However, it takes efforts to choose weight function. Document [12, 13] proposed a design method of H_∞ control theory to obtain a stable compensator to improve its robustness, but compensators with good performance are often high-end and complex to implement. Most of the repetitive control design techniques mentioned in the literature are based on the transfer function method, which is not sufficient to deal with the time-varying uncertainty caused by the load change. Not only is the solution process complicated, but also the steady-state error and phase angle margin are used to judge the system performance after consulting the benchmark amount, which is often determined through heuristics.

Deadbeat control is widely studied in terms of its fast response speed and easy implementation, rather than its

poor robustness [13]. Document [14] proposes an improved deadbeat control strategy to compensate for the delay caused by the digital implementation by predicting the system behavior, but it needs to select large orders to ensure the accuracy of the algorithm prediction, which makes the calculation process complicated.

Inspired by the above literature, in order to reduce the harmonic content of the PCC point voltage of the z-source inverter with nonlinear loads and achieve power sharing, this paper proposes a H_∞ state feedback deadbeat repetitive control based on a parameter-optimized droop controller (H_∞SFDBRC) strategy. Aiming at the disadvantages of traditional repetitive control being insufficient to deal with the time-varying uncertainty caused by load changes, the complexity of the solution process, and the difficulty of verifying conditions, this paper designs an improved repetitive control system that optimizes the design of the repetitive controller, constructs a state space expression formula, and introduces H_∞ state feedback control, and thus the design of the repetitive controller is transformed into a convex optimization problem with a set of LMI constraints according to the Lyapunov functional; due to the inherent delay characteristics of repetitive control, in order to improve the transient performance of the system to obtain better antidisturbance characteristics, a new conceptual topology is proposed, with the deadbeat control technology, to provide fast dynamic response after the start of the system or during the large load step changes. A simulation model was built on the MATLAB/Simulink simulation platform, and the simulation results verified the effectiveness of the control strategy.

2. System Structure

The single-phase off-grid inverter system based on z-source inverter is shown in Figure 1. The designed control system has three constituent modules: repeated control, deadbeat control, and droop control modules. L_f and C_f are the filter inductors and capacitors, R_{Lf} is an equivalent series resistance of the filter inductance, R_i and L_i are the line impedance, I_{Labc} is an inductor current, u_C is the capacitor voltage, V_{0abc} is the output voltage, K_{11}, K_{12}, and K_2 are competitive controller coefficients calculated by LMI, $Q(s)$ $e^{-s\tau}$ is a first-order low-pass filter, K_3, K_4, K_5, and K_6 are the coefficients of the deadbeat control system, u_k is the SVPWM voltage modulation signal, k_p and k_q are droop coefficients, f_0 is the rated frequency, U_0 is the voltage amplitude when output reactive power is zero, and P, Q are measured values of active and reactive power, respectively. The measured inverter output voltage V_{0abc} and filter inductor current I_{Labc} are converted, by Park Formation, into voltage V_{0abc} and current I_{Labc} on the two-phase rotating coordinates. The input of voltage V_{0dq} and current I_{Labc} into the droop control system generates reference voltage r, which is then converted into modulate wave signals under the repetitive and deadbeat control. After controlling the inverter's on-off switch pipe, the effective tracking of the voltage signal can be realized.

Figure 1 shows the simplified topology of the droop control outer loop and the voltage control inner loop (Figure 2).

The z-source inverter consists of two capacitors and two inductors to form an X-type network, which connects the DC source with the three-phase inverter. The function of the diode is to prevent the current from flowing back to the DC side. In normal operation, there are two states of straight-through and nonstraight-through; when the upper and lower power devices of the same bridge arm are turned on at the same time, it is in the straight-through state, through which the z-source inverter can flexibly boost and step down; the nonstraight-through state refers to the traditional inverter state. Figures 3 and 4 are equivalent circuits in both straight-through and nonstraight-through states, respectively.

Since the three-phase filter circuit parameters are consistent and the d_q-axis is independent after the coordinates are transformed, simply analyze the d-axis single-phase LC filter as shown in Figure 5; the DC voltage, z-source network, and inverter are equivalent to a voltage source $u(t)$, where the influence of both the linear and nonlinear loads on the controlled output voltage is modeled by the uncertain load admittance $Y_0(t)$ and the external current source $i_d(t)$.

Select inductor current I_{Ld} and capacitor voltage V_{0d} as state variables to establish circuit equations according to Kirchhoff's law

$$\begin{cases} u(t) = L_f \dfrac{\mathrm{d}I_{Ld}(t)}{\mathrm{d}t} + R_{Lf} I_{Ld}(t) + V_{0d}(t), \\ I_{Ld}(t) = C_f \dfrac{\mathrm{d}V_{0\,d}(t)}{\mathrm{d}t} + V_{0d}(t) Y_0(t) + i_d(t). \end{cases} \tag{1}$$

Organize it into a matrix form

$$\begin{bmatrix} \dfrac{\mathrm{d}I_{Ld}(t)}{\mathrm{d}t} \\ \dfrac{\mathrm{d}V_{0d}(t)}{\mathrm{d}t} \end{bmatrix} = \begin{bmatrix} -\dfrac{R_{Lf}}{L_f} & -\dfrac{1}{L_f} \\ \dfrac{1}{C_f} & \dfrac{Y_0(t)}{C_f} \end{bmatrix} + \begin{bmatrix} \dfrac{1}{L_f} \\ 0 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ \dfrac{1}{C_f} \end{bmatrix} i_d(t),$$

$$V_{0d}(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} I_{Ld}(t) \\ V_{0d}(t) \end{bmatrix}. \tag{2}$$

If $x(t) = [I_{Ld}(t)\ V_{0d}(t)]^T$, it can be written as a state space expression.

$$\begin{cases} \dot{x}(t) = \mathbf{A}(Y_0(t))\mathbf{x}(t) + \mathbf{B}u(t) + \mathbf{B}_d i_d(t), \\ y(t) = \mathbf{C}\mathbf{x}(t), \end{cases} \tag{3}$$

where $x(t) \in R$ is the state vector of the inverter, $u(t) \in R$ is control input, $y(t) \in R$ is control output, $i_d(t) \in R$ is periodic interference, and $A(Y_0(t))$ is the matrix function of the uncertain parameter $Y_0(t)$.

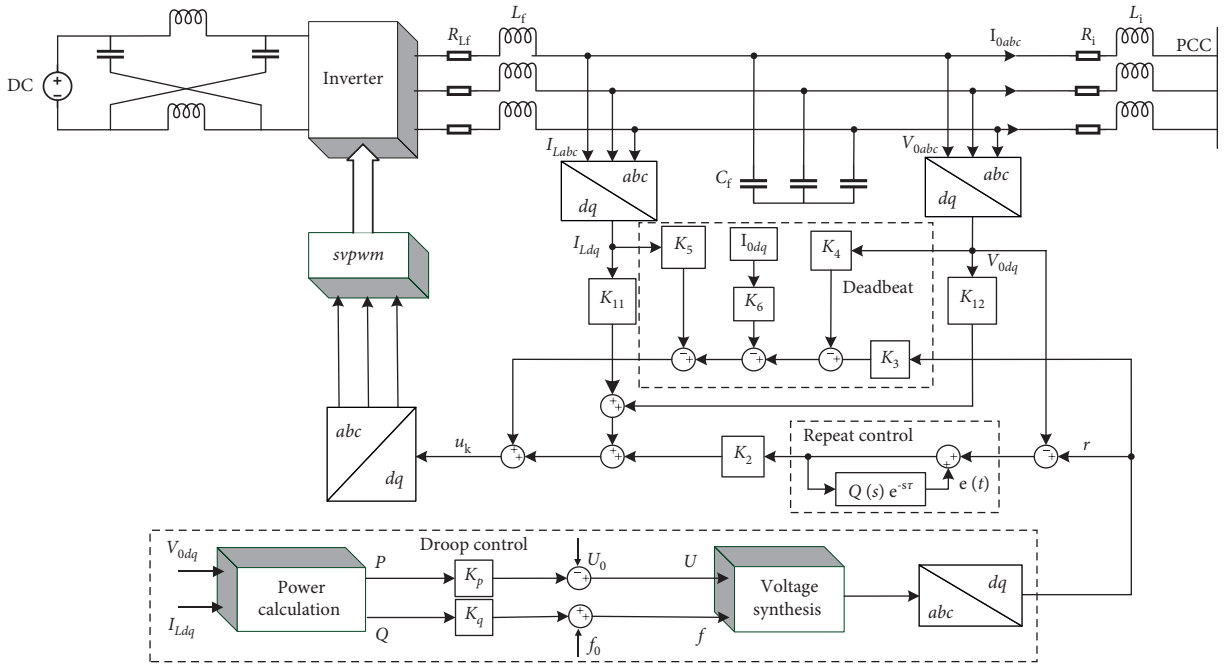


Figure 1: Control block diagram of island microgrid inverter.

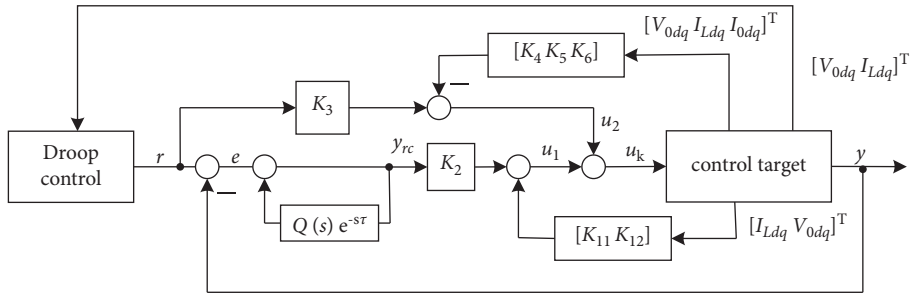


Figure 2: System simplified control topology diagram.

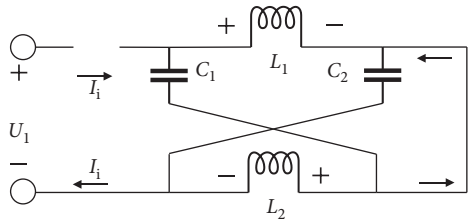


Figure 3: Straight-through commutation state.

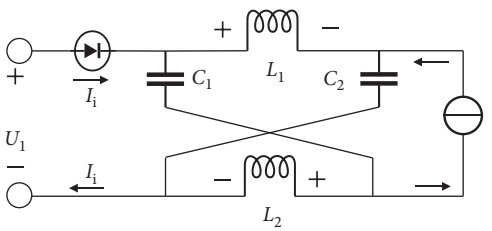


Figure 4: Nonstraight-through commutation state.

Suppose that the minimum and maximum values of the $Y_0(t)$ are known

$$Y_{\min} \le Y_0(t) \le Y_{\max}. \tag{4}$$

The parameter $Y_0(t)$ is usually converted according to its nominal value Y_N and the deviation Y_D, as follows:

$$Y_0(t) = Y_N + \delta(t) Y_d, \quad \delta(t) \in [-1, 1], \tag{5}$$

where $Y_N = Y_{\min} + Y_{\max}/2, Y_d = Y_{\min} - Y_{\max}/2$.

According to (5)

$$\mathbf{A}(Y_0(t)) = \mathbf{A}(Y_N) + \mathbf{H}(Y_d)\delta(t)\mathbf{E}, \quad \delta(t) \in [-1, 1], \tag{6}$$

where $A(Y_N)$, $H(Y_D)$, and E are the constant matrix of the uncertain structures, as given by the following formula:

$$\begin{aligned} \mathbf{A}(Y_N) &= \begin{bmatrix} -\dfrac{R_{Lf}}{L_f} & -\dfrac{1}{L_f} \\ \dfrac{1}{C_f} & -\dfrac{Y_N}{C_f} \end{bmatrix}, \\ \mathbf{H}(Y_d) &= \begin{bmatrix} 0 & 0 \\ 0 & -\dfrac{Y_d}{C_f} \end{bmatrix}, \\ \mathbf{E} &= I_2. \end{aligned} \tag{7}$$

2.1. Voltage Control Strategy. Repetitive control is to reflect the deviation from the previous operation to the present and add it to the controlled object together with the "current deviation" for control to improve tracking accuracy and suppress periodic interference. However, such closed-loop systems have infinite poles on the virtual axis. It is impossible to achieve stability through classical control methods.

Figure 6 shows a repetitive controller with a low-pass filter; set the low-pass filter $Q(s)$ as

$$\begin{aligned} Q(s) &= \frac{\omega_c}{s + \omega_c}, \\ \omega_c &= \frac{1}{T}, \end{aligned} \tag{8}$$

where ω_c and T are the corner frequency and time constants of the first-order low-pass filters, respectively. The method of determining the size of ω_c is as follows:

$$\begin{cases} |q(j\omega)| \approx 1, & \omega \ll \omega_0, \\ |q(j\omega)| < 1, & \omega > \omega_0, \end{cases} \tag{9}$$

where ω_0 is the frequency bandwidth of the modulation reference signal: $\omega_0 = 2\pi/\tau$.

From Figure 6, the transfer function of the repeated controller is

$$G_{rc}(s) = \frac{1}{1 - Q(s)e^{-s\tau}}. \tag{10}$$

Deduced from formulas (8) and (9), the state space of formula (9) is available

$$\begin{cases} \dot{x}_{rc}(t) = -\omega_c x_{rc}(t) + \omega_c x_{rc}(t - \tau) + \omega_c e(t - \tau), \\ y_{rc}(t) = x_{rc}(t) + e(t), \end{cases} \tag{11}$$

where $x_{rc}(t)$ is a low-pass filter state variable.

2.2. Design of H_∞ State Feedback Repetitive Controller. Construct the augmentation vector as follows:

$$\mathbf{Z}(t) = \begin{bmatrix} \mathbf{x}(t) \\ x_{rc}(t) \end{bmatrix} \in R^{(n+1)}. \tag{12}$$

Rules (3) and (11) can be converted to

$$\dot{Z}(t) = (\mathbf{A}_a + \Delta\mathbf{A}_a(t))\mathbf{Z}(t) + \mathbf{A}_d\mathbf{Z}(t - \tau) + \mathbf{B}_a u(t) + \mathbf{B}_q\mathbf{q}(t), \tag{13}$$

where

$$\begin{aligned} \mathbf{q}(t) &= [\, r(t) \;\; i_d(t) \,]' \in R^2, \\ \mathbf{A}_a &= \begin{bmatrix} \mathbf{A}(Y_N) & 0_{2\times1} \\ 0_{1\times2} & -\omega_c \end{bmatrix}, \\ \mathbf{B}_a &= \begin{bmatrix} \mathbf{B} \\ 0 \end{bmatrix}, \\ \mathbf{A}_d &= \begin{bmatrix} 0_{1\times2} & 0 \\ -\mathbf{C}\omega_c & \omega_c \end{bmatrix}, \\ \Delta\mathbf{A}_a(t) &= \mathbf{H}_a\delta(t)\mathbf{E}_a \\ \mathbf{H}_a &= \begin{bmatrix} \mathbf{H}(Y_d) \\ 0_{1\times2} \end{bmatrix}, \\ \mathbf{E}_a &= \begin{bmatrix} E' \\ 0'_{1\times2} \end{bmatrix}', \\ \mathbf{B}_q &= \begin{bmatrix} 0_{2\times1} & \mathbf{B}_d \\ \omega_c & 0 \end{bmatrix}. \end{aligned} \tag{14}$$

As shown from Figure 7, the control rate in the closed-loop system (13) is

$$u_1(t) = \mathbf{K}_1 x(t) + K_2 y_{rc}(t), \tag{15}$$

where $\mathbf{K}_1 = [\, K_{11} \;\; K_{12} \,]$.

Override the $u_1(t)$

$$u_1(t) = \mathbf{F}\mathbf{Z}(t) + K_2 r(t), \tag{16}$$

where $\mathbf{F} \in R^{1\times(n+1)} = [(\mathbf{K}_1 - K_2\mathbf{C})\, K_2]$.

Then the augmented system (13) is rewritten as

$$\dot{Z}(t) = (\mathbf{A}_\Delta + \mathbf{B}_a\mathbf{F})\mathbf{Z}(t) + \mathbf{A}_0\mathbf{Z}(t - \tau) + \overline{\mathbf{B}}_q\mathbf{q}(t), \tag{17}$$

where $A_\Delta = \mathbf{A}_a + \Delta\mathbf{A}_a(t)$, and $\overline{\mathbf{B}}_q$ depends on K_2

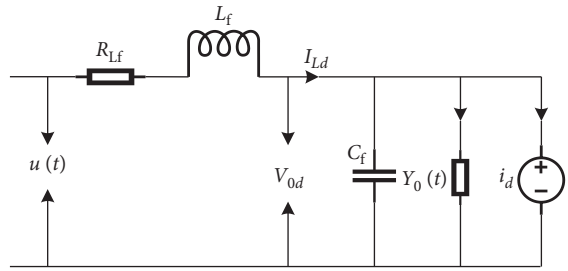


Figure 5: LC filter single-phase equivalent circuit.

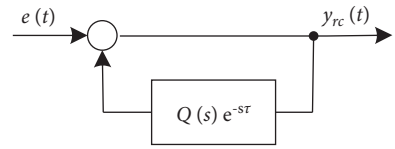


Figure 6: Low-pass filter repetitive control topology.

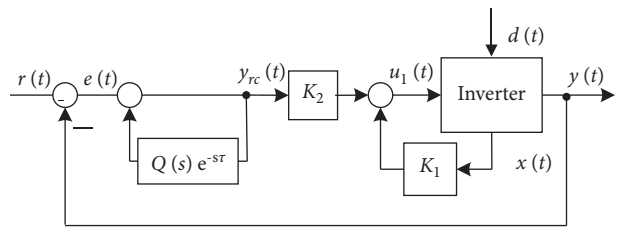


Figure 7: Repetitive control topology.

$$\overline{\mathbf{B}}_q = \begin{bmatrix} \mathbf{B}K_2 & \mathbf{B}_d \\ \omega_c & 0 \end{bmatrix}. \tag{18}$$

To validate the stability of the above system, ignoring the external input $q(t)$, the closed-loop system can be abbreviated as

$$\dot{Z}(t) = (\mathbf{A}_\Delta + \mathbf{B}_a\mathbf{F})\mathbf{Z}(t) + \mathbf{A}_d\mathbf{Z}(t-\tau). \tag{19}$$

Then for a given cutoff frequency ω_c, the gain F needs to be determined so that the system is asymptotically stable for any $Y_0(t)$.

In view of the above augmented system, performance criteria are introduced:

$$J(p(t)) \coloneqq \|p(t)\|_2^2 = \int_0^\infty p(t)'p(t)\mathrm{d}t, \tag{20}$$

where $p(t)$ defines the performance output for

$$p(t) \coloneqq \mathbf{C}_p z(t) + \mathbf{D}_p u(t), \tag{21}$$

where C_p and D_p are the constant matrix of the appropriate dimensions.

Further ensure that the system trajectory has a given exponential decay rate of α

$$\|\mathbf{Z}(t)\| \le \beta\|\mathbf{Z}(0)\|e^{-\alpha t}, \quad t > 0, \tag{22}$$

where β is a positive scalar and $Z(0)$ is the initial state.

$$\mathbf{Z}(\phi) = 0, \quad \forall\phi \in [-\tau, 0). \tag{23}$$

In order to obtain a sufficient condition for the robust asymptotic stability of system (19), according to [16], the above problem can be solved by the following lemma.

Lemma 1 (see [15]). *For the given positive scalars ω_c and α, consider formulas (17) and (21). Suppose there is a symmetric positive definite matrix W, $S \in R^{3\times3}$, matrix $Y \in R^{1\times3}$, and positive scalars λ and ν are satisfied with*

$$\begin{bmatrix} \Gamma(\mathbf{W},\mathbf{S},\upsilon) & e^{\alpha t}\mathbf{A}_d\mathbf{W} & \mathbf{W}\mathbf{E}_a & \mathbf{W}\mathbf{C}_p' + Y'\mathrm{D}_p' \\ e^{\alpha t}\mathbf{W}A'_{\mathrm{d}} & -\mathbf{S} & 0_{3\times3} & 0_{3\times1} \\ E_a'\mathbf{W} & 0_{3\times3} & -\upsilon\mathbf{I}_3 & 0_{3\times1} \\ \mathbf{C}_p\mathbf{W} + \mathbf{D}_p\mathbf{Y} & 0_{1\times3} & 0_{1\times3} & -\lambda \end{bmatrix} < 0, \tag{24}$$

where $\Gamma(\mathbf{W},\mathbf{S},\upsilon) = \mathbf{A}_a\mathbf{W} + \mathbf{W}\mathbf{A}_a' + 2\alpha\mathbf{W} + \mathbf{B}_a\mathbf{Y} + Y'\mathbf{B}_a' + \mathbf{S} + \upsilon\mathbf{H}_a\mathbf{H}_a'$.

Then the closed-loop system in equation (17) is gradually stable when the gain $\mathbf{F} = \mathbf{Y}\mathbf{W}^{-1} = [\,\mathbf{F}_1 \;\; \mathbf{F}_2\,]$.

Note 1. Lemma 1 gives the stability conditions of H_∞ repetitive control when the inverter carries an uncertain load and the design method of the state feedback control rate. It shows that the closed-loop system (19) is progressively stable, so the closed-loop system (17) is internally stable.

In this paper, LMI is used to optimize the design of the repetitive controller, and the MATLAB toolbox is used to solve it. Compared with the traditional repetitive controller design method that determines, by heuristics, reference quantity, like steady-state error and phase angle margin, this paper acquires them by solving the linear matrix inequality and at the meantime simplifies the design process by reducing the number of filters and the cost of the whole system, facilitating the solution of the time-varying uncertainty caused by load change. See Table 1 for the inverter parameters involved.

Take $\omega_c = 1000$ and $\alpha = 155$, and the parameters of the repetitive controller obtained by using LMI according to Lemma 1 are

$$\begin{aligned} \mathbf{K}_1 &= \left[-181.2753 \quad -153.7798 \right], \\ K_2 &= 5.2749 \times 10^3, \end{aligned} \tag{25}$$

2.3. H_∞ State Feedback Deadbeat Repetitive Control. Due to the inherent nature of the delay characteristic of the repetitive controller, the dynamic performance of the system is poor. While deadbeat control has the advantages of quick instantaneous reaction and low harmonic distortion rate, in order to improve the dynamic performance of the system, compensate for the distortion caused by the dead time of the switch, introduce deadbeat control technology, and propose a new conceptual topology as shown in Figure 8.

The sampling principle of deadbeat control is shown in Figure 9, the sampling period is expressed as T, the output value of the inverter is expressed as $+E$ and $-E$, $\Delta T(K)$ represents the adjusting width of the K th cycle square wave, and the same $\Delta T(K)$ represents the width of the $K+1$ th cycle. The voltage value in the sampling period is determined by the sampling value at that time and the reference value at the next time.

When the microgrid inverter is working, the introduced deadbeat control technology takes into account the effect of the actual load current, so that the entire system can automatically compensate for load disturbances during the transient performance, so that it can be used during system startup or load step. It provides fast dynamic response during the change period.

The system adopts discrete-time simulation. The reason why the digital system can achieve the deadbeat control effect is that the output of the next beat of the system can always be expressed as a linear combination of the current input control quantity and the system state variable. When the system deviates from the reference value, it will respond quickly. The load disturbance is compensated and the pulse width of the next switching period is calculated. According to the state space expression of formula (3), it is to discretize on the equivalent impulse principle; since the deadbeat controller is mainly used to provide rapid dynamic response during system startup or load step responses, to simplify the calculation, taking $Y_0(t)$ as a fixed value is available to

$$\begin{cases} x(k+1) = \mathbf{G}x(k) + \mathbf{M}_1 u_2(k) + \mathbf{M}_2 I_{0d}(k), \\ y(k) = \mathbf{C}x(k), \end{cases} \tag{26}$$

where I_{0d} is the d-axis load current.

$$\begin{aligned} \mathbf{G} &= e^{\mathbf{A}(Y_0)T_s} = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix}, \\ \mathbf{x}(k) &= \begin{bmatrix} V_{0d}(k) \\ I_{Ld}(k) \end{bmatrix}, \\ \mathbf{M}_1 &= \mathbf{A}(Y_0)^{-1}\left(e^{\mathbf{A}(Y_0)T_s} - \mathbf{I}\right)\mathbf{B} = \left[m_{11} \quad m_{12} \right]^T, \\ \mathbf{M}_2 &= \mathbf{A}(Y_0)^{-1}\left(e^{\mathbf{A}(Y_0)T_s} - \mathbf{I}\right)\mathbf{B}_d = \left[m_{21} \quad m_{22} \right]^T. \end{aligned} \tag{27}$$

Expanded by formula (26)

$$\begin{cases} V_{0d}(k+1) = g_{11}V_{0d}(k) + g_{12}I_{Ld}(k) + m_{11}u_2(k) + m_{21}I_{0d}(k), \\ I_{Ld}(k+1) = g_{21}V_{0d}(k) + g_{22}I_{Ld}(k) + m_{12}u_2(k) + m_{22}I_{0d}(k). \end{cases} \tag{28}$$

According to formula (28), the capacitor voltage $V_{0d}(k+1)$ at t_{k+1} time is determined by $V_{0d}(k)$, $I_{Ld}(k)$, and $u_2(k)$ at t time. Conversely, if $V_{0d}(k+1)$, $V_{0d}(k)$, and $I_{Ld}(k)$ are known at t_k time, then the output voltage $u_2(k)$ at t_k time can be calculated. The calculation formula can be derived from formula (28):

$$u_2(k) = \frac{1}{m_{11}}\left[V_{0d}(k+1) - g_{11}V_{0d}(k) - g_{12}I_{Ld}(k) - m_{21}I_{0d}(k)\right]. \tag{29}$$

In fact, both $V_{0d}(k)$ and $I_{Ld}(k)$ are t_k time sampling values, which are known. Now let us determine the capacitor voltage $V_{0d}(k+1)$ at the time of t_{k+1}. It can be seen from formula (28) that the output of the system is the capacitor voltage and the ideal output voltage of the inverter is the standard sinusoidal reference voltage. Therefore, the reference voltage $r(k+1)$ of t_{k+1} can be used instead of the capacitor voltage $V_{0d}(k+1)$ of t_{k+1}, that is,

$$\begin{aligned} u_2(k) &= \frac{1}{m_{11}}\left[r(k+1) - g_{11}V_{0d}(k) - g_{12}I_{Ld}(k) - m_{21}I_{0d}(k)\right] \\ &= K_3 r(k+1) - K_4 V_{0d}(k) - K_5 I_{Ld}(k) - K_6 I_{0d}(k). \end{aligned} \tag{30}$$

System integrated control law $u_k = u_1 + u_2$; therefore, it is necessary to obtain the discrete-time model of u_1, where $K_1 x(t)$ is obtained from static feedback and does not require a discretization process. $K_2 y_{rc}(t)$ is derived from the dynamic compensator and must be discretized to determine the relationship between $y_{rc}(t)$ and $e(t)$. Discretization of equation (10) can obtain the discrete-time transfer function from $e(z)$ to $y_{rc}(z)$

$$G_{rc}(z) = \frac{(2+\omega_t) + (\omega_t - 2)z^{-1}}{(2+\omega_t) + (\omega_t - 2)z^{-1} - \omega_t z^{-\gamma} - \omega_t z^{-\gamma-1}}, \tag{31}$$

where $\omega_t = \omega_c T_s$, in terms of difference equations

Table 1: Inverter parameters.

Parameter	Value
Filter inductance L_f/mH	0.6
Filter capacitor C_f/μF	1500
Damping resistance R_L/Ω	0.01
Minimum admittance/S	0.0001
Maximum admittance/S	0.2
Switching frequency f/KHz	21.6
DC bus voltage/V	800

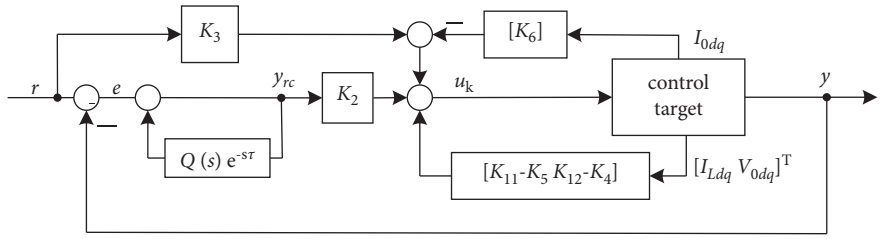


Figure 8: H_∞ state feedback repeated deadbeat control topology.

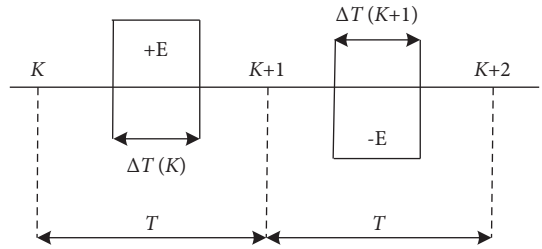


Figure 9: Structure diagram of deadbeat control sampling.

$$y_{rc}(k) = e(k) + \frac{-2+\omega_t}{2+\omega_t}\left[e(k-1) - y_{rc}(k-1)\right] + \frac{\omega_t}{2+\omega_t}\left[y_{rc}(k-\gamma) + y_{rc}(k-\gamma-1)\right]. \tag{32}$$

From equations (31) and (32),

$$u_1(k) = K_{11}I_{Ldq}(k) + K_{12}V_{0dq}(k) + K_2 y_{rc}(k). \tag{33}$$

The system control law is

$$u_k(k) = (K_{11}-K_5)I_{Ldq}(k) + (K_{12}-K_4)V_{0dq}(k) + K_2 y_{rc}(k) + K_3 r(k+1) - K_6 I_{0dq}(k). \tag{34}$$

According to the conclusion of the document [17], when the H_∞ state feedback repetitive control and deadbeat control act independently, the system is stable, so the composite system is also stable.

In summary, the design steps of the control law in this paper are as follows.

Step 1. Give suitable ω_c and α.

Step 2. Solve the following convex optimization problem:

$$\min_{W,Y,S,v,\lambda} \lambda \text{ Satisfaction.} \tag{35}$$

Step 3. Compute

$$\mathbf{F} = \mathbf{YW}^{-1} = [\,\mathbf{F}_1\ \ \mathbf{F}_2\,], \quad K_1 = F_1 + K_2 C, \quad K_2 = F_2. \tag{36}$$

Step 4. Solve the output voltage u_2 at $t = k$, and obtain the state feedback coefficient.

2.4. Droop Control. This paper studies the island mode of the inverter; the calculation formula of average active power P and average reactive power Q is as follows:

$$\begin{cases} P = \left(V_{0d}I_{Ld} + V_{0q}I_{Lq}\right)\dfrac{w_{f1}}{S+w_{f1}}, \\ Q = \left(V_{0d}I_{Lq} - V_{0q}I_{Ld}\right)\dfrac{w_{f2}}{S+w_{f2}}. \end{cases} \tag{37}$$

In the formula, w_{f1} and w_{f2} are the cutoff frequency of the low-pass filter.

The droop control equation is

$$\begin{cases} f = f_0 + k_q Q, \\ U = U_0 - k_p P. \end{cases} \tag{38}$$

In the formula, k_q and k_p are droop characteristic coefficients, f_0 is rated frequency, and U_0 is voltage amplitude when output reactive power is 0. The control parameters are shown in Table 2.

2.5. Simulation Verification. This paper establishes a microgrid operation simulation model with two DG on the MATLAB/Simulink software platform as shown in Figure 10. The DC bus voltage of DG module is maintained by ideal power source, which adopts the same LC filter and line impedance. The standard feeder impedance of low-voltage microgrid is $0.642 + j0.0083\ \Omega$/km. The line impedance seen from DG1 and DG2 is $1.284 + j0.0166\ \Omega$/km. The common load load3 is connected to the common AC bus of the microgrid and through on and off switch K.

In order to verify the effectiveness of the control strategy proposed in this paper, the following simulation experiments are carried out. The load is a rectifier with a $21.8\ \Omega$ output in parallel with a set of $10.9\ \Omega$ resistors.

Figure 11(a) shows the phase A output voltage during system startup. It can be seen that, due to the deadbeat control technology, the voltage has achieved good tracking in a very short time. As shown in Figure 11(b), the periodic error signal begins to fluctuate and then quickly converges to a very small level (less than 0.5 V), and the entire system is stable. Then apply a large load step change test to check the system control performance. At the 0.4 s time point, the nonlinear load suddenly increases from one group to two groups. Figure 12(a) shows the corresponding PCC point voltage and current changes. It can be seen that, for large load step changes, the system is robust and withstands variation. Figure 12(b) shows the changes of active power and reactive power when the system load changes step by step. It can be seen that the system can still maintain a stable and equal distribution of active and reactive power when the load step changes, and the power fluctuates in it tending to be stable in a short period of time, indicating that the proposed control strategy has good anti-interference performance and dynamic response performance.

According to the comparison method of [18], the proposed H_∞ state feedback deadbeat repetitive control is compared with H_∞ repetitive control and PI control (hereinafter referred to as H_∞SFDBRC, H_∞RC, and PI), in which H_∞ repetitive control and PI control are designed by ourselves, and other conditions such as inverter parameters and line impedance are consistent. It is proved that the designed control strategy can improve the performance of inverter PCC point voltage THD under nonlinear load.

When using PI control, the PCC point voltage waveform, voltage error, and spectrum analysis results are shown in Figure 13(a). It can be seen from the figure that the PCC point voltage waveform has obvious distortion, the steady-state error is the largest, and the tracking performance is poor. The quality of the grid-connected voltage waveform is poor, and the voltage harmonic content is 4.91%.

Table 2: Droop control parameters.

Parameter	Value
Line impedance/Ω	$1.284 + j0.0166$
Filter parameters	$w_{f1} = 50$; $w_{f2} = 100$
Rated frequency/Hz	50
Voltage amplitude/V	110
Droop coefficient	$k_p = 10^{-5}$, $k_q = 3 \times 10^{-4}$

When using H_∞RC, the PCC point voltage waveform, voltage error, and spectrum analysis results are shown in Figure 13(b). The introduction to the internal model link enables the system to better compensate for the harmonic voltage, and the PCC point voltage waveform quality is better than PI control strategy, the voltage waveform is smoother, the harmonic compensation effect is good, the voltage harmonic content is 2.44%, and the steady-state error is better than PI control, but the dynamic response speed is slow.

When using H_∞SFDBRC, the PCC point voltage waveform, voltage error, and spectrum analysis results are shown in Figure 13(c). At this time, the system steady-state error is the smallest, and the excellent tracking performance significantly improves the voltage waveform quality. The deadbeat control technology with the introduction to the system enhances the anti-interference performance of the system, enabling the system to quickly respond to various sudden problems. The quality of the voltage waveform is better than the first two control strategies, and the harmonic compensation effect is very good, and the voltage harmonic content is only 0.67%.

In order to further verify the feasibility of the proposed scheme, the PCC point voltage waveform and power changes in the following two cases are simulated. Under the situation, only the inverter DG1 runs when the system starts, and the DG2 switch is closed at 0.4 s, and the simulation time is 1s. The simulated waveforms are shown in Figures 14(a)–14(c). When the microgrid is operating normally, it is connected to DG2 for interconnection at 0.4 s. DG1 and DG2 maintain a good coordinated operation. The PCC voltage quickly returns to a stable sinusoidal curve after slight fluctuations, during which active power and reactive power can also move quickly to achieve power sharing and maintain stability. It can also move quickly to achieve power sharing and maintain stability.

In case two, the two inverters run in parallel, the simulation time is 1 s, the switch K_n is disconnected at 0.4 s, and DG2 exits operation. From the simulation waveform diagrams 14(d)–14(f), it can be seen that when the microgrid is running in parallel, when one of the DGs quits operation due to a fault, the microgrid can react quickly and reach a new stable state, where the PCC voltage is almost invariant and remains as a sine curve, and the distribution and transformation of active power and reactive power in this process also maintain extremely high accuracy.

In summary, the control strategy proposed in this article can still ensure the stability and normal operation of the microgrid when the DG is connected or disconnected or the load changes.

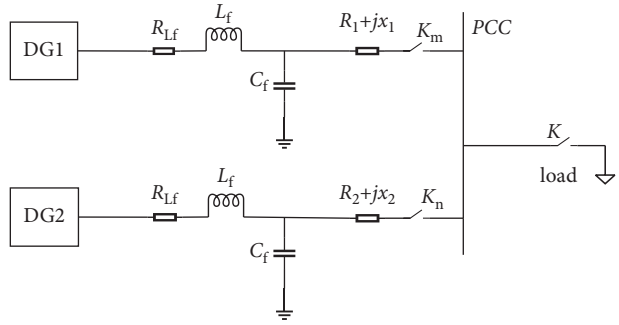


Figure 10: Microgrid simulation model.

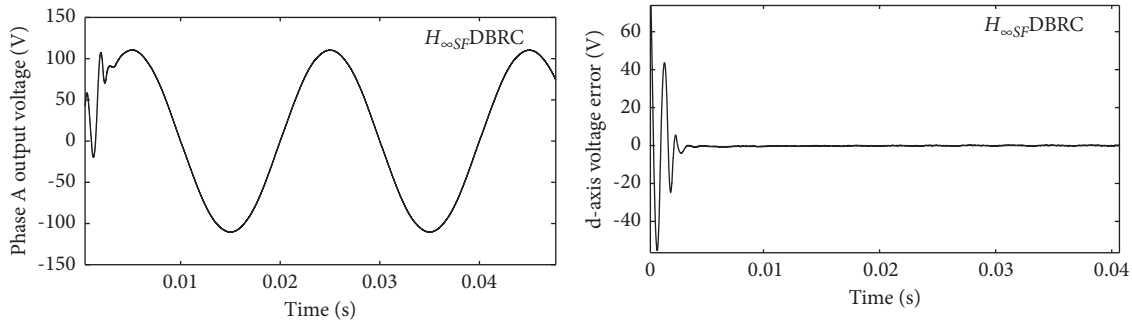


Figure 11: Output voltage of phase A during system startup and voltage error signal of d-axis seen by the repetitive controller. (a) Phase A output voltage. (b) d-axis voltage error obtained.

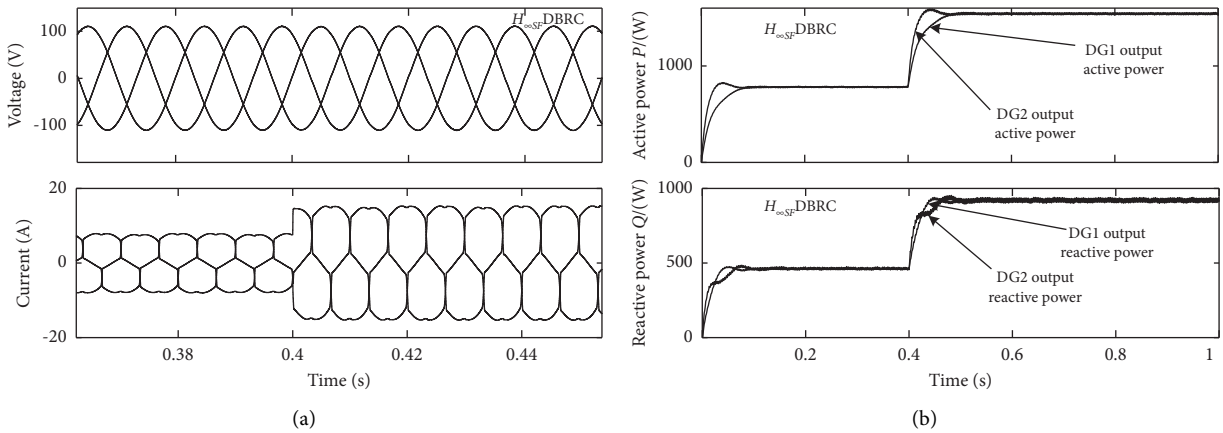


Figure 12: Three-phase output voltage during steady-state operation and three-phase voltage and current during load step changes. (a) PCC voltage and current. (b) Active and reactive power.

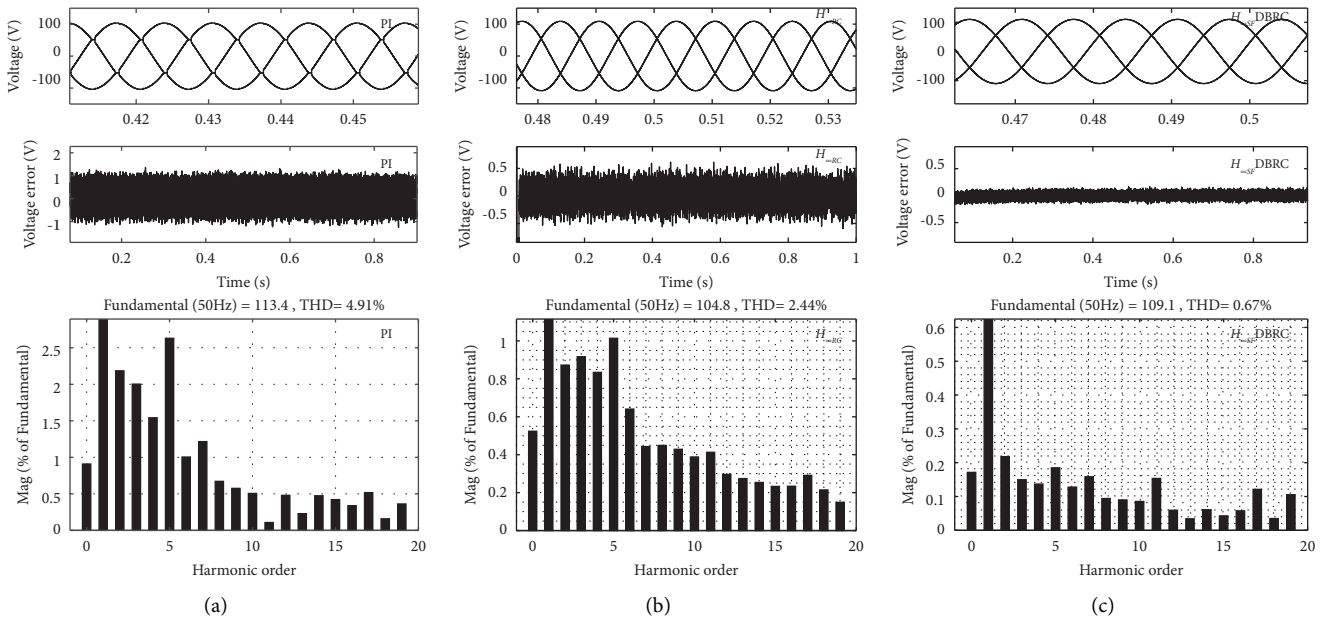


Figure 13: PCC point voltage, d-axis voltage error, and its THD value. (a) PI controller. (b) H_∞RC controller. (c) H_∞SFDBRC controller.

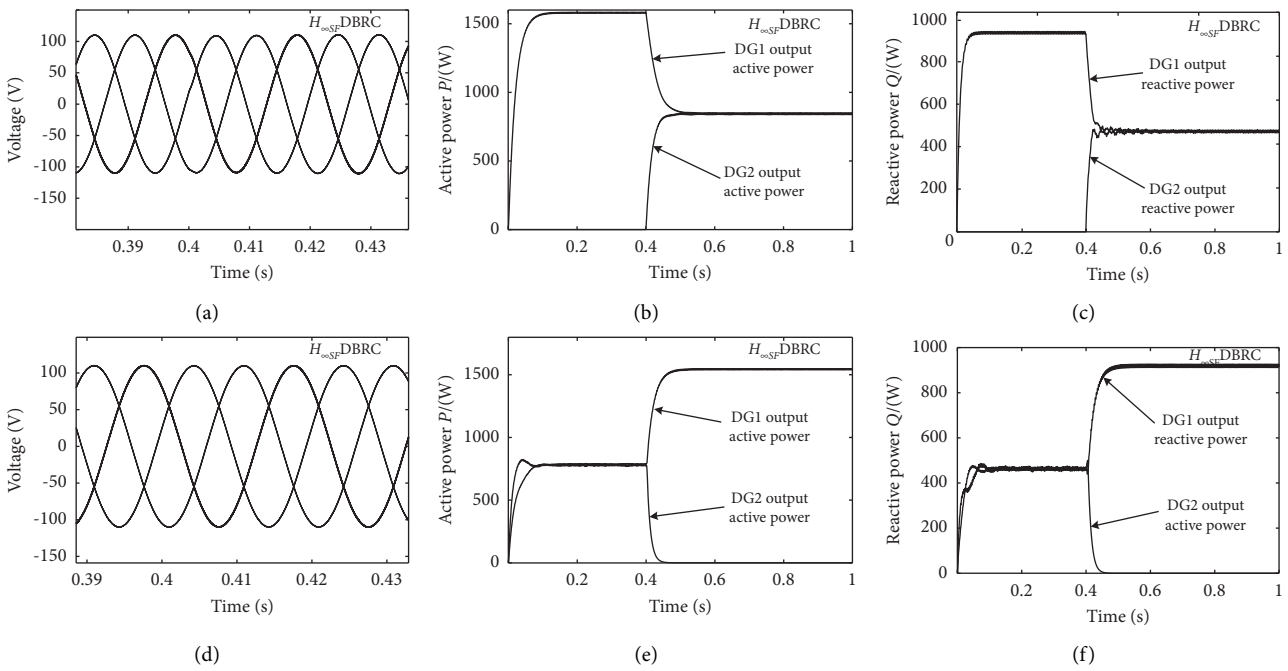


Figure 14: PCC voltage, DG output active, and reactive power. (a) PCC voltage waveform. (b) DG output active power. (c) DG output reactive power. (d) PCC voltage waveform. (e) DG output active power. (f) DG output reactive power.

3. Conclusion

Aiming at the problem of voltage distortion at the public grid connection point of island microgrid caused by nonlinear load, this paper designs a H_∞ state feedback deadbeat repetitive control strategy to reduce the total harmonic distortion of the output voltage. Through theoretical analysis and research, the following conclusions can be drawn.

(1) Lyapunov functional is used to optimize the design problem of repetitive controllers, the robustness of the closed-loop system is ensured by introducing state feedback, combined with H_∞ control theory, the design problem is transformed into a set of linear matrix inequality constraints' convex optimization problem, which can simplify the repetitive control design, and compared with the traditional design

method, the parameters representing the stability of the system can be accurately obtained and verified. It does not need to be obtained through repeated tests by trial method. It is more suitable for practical engineering applications and has the advantages of good stability, low harmonic content, and fast convergence speed.

(2) The introduction of deadbeat control technology not only effectively improves the response speed of the system, but also compensates for the distortion introduced by the switch dead time. Through simulation, we can find that the microgrid can quickly react and respond regardless of the actual system startup period or the large load step change period. A stable state is reached.

(3) The simulation results show that, compared with PI control and H_∞ repetitive control, the proposed control strategy can effectively reduce harmonics, improve the voltage quality at PCC point, and has good control performance under nonlinear load conditions.

Data Availability

The data used to support the findings of this study are included within the article. See Table 1 for the inverter parameters involved: parameters and values: z-source capacitor, C/μF, 4000, z-source inductor, L/mH, 0.5, filter inductor, Lf/mH, 0.6, filter capacitor, Cf/μF, 1500, damping resistance, RL/Ω 0.01, minimum admittance/S, 0.0001, maximum admittance/S, 0.2, switching frequency, f/KHz, 21.6, and DC bus voltage/V, 800. Take $\omega_c = 1000$ and $\alpha = 155$. The control parameters are shown in Table 2: parameters and value: line impedance/Ω $1.284 + j0.0166$, filter parameters $w_{f1} = 50$; $w_{f2} = 100$, rated frequency/Hz 50, voltage amplitude/V 110, and droop coefficient. $k_p = 10 - 5$ and $k_q = 3 \times 10 - 4$.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] M. A. Hossain, H. R. Pota, M. J. Hossain, and F. Blaabjerg, "Evolution of microgrids with converter-interfaced generations: challenges and opportunities," *International Journal of Electrical Power & Energy Systems*, vol. 109, no. JUL, pp. 160–186, 2019.

[2] D. Petreus, R. Etz, T. Patarau, and M. Cirstea, "An islanded microgrid energy management controller validated by using hardware-in-the-loop emulators," *International Journal of Electrical Power & Energy Systems*, vol. 106, no. MAR., pp. 346–357, 2019.

[3] P. Sreedharan, J. Farbes, E. Cutter, C. K. Woo, and J. Wang, "Microgrid and renewable generation integration: university of California, San Diego," *Applied Energy*, vol. 169, no. MAY1, pp. 709–720, 2016.

[4] C. Meng, X. X. Ning, and B. Z. Lai, "Hierarchical frequency control strategy of microgrids with virtual synchronous generators," *High Voltage Engineering*, vol. 44, no. 4, pp. 1278–1284, 2018.

[5] Z. Li, Z. Jiang, J. Huang, W. Ma, and Z. Huang, "Time-frequency voltage control strategy of microgrid inverter based on linear active disturbance rejection control," *Automation of Electric Power Systems*, vol. 44, no. 10, pp. 145–154, 2020.

[6] D. Li, B. Wang, J. Liu, and W. Xu, "Research on droop control method with the compounding feed forward compensation for micro-grid inverter," *Power Electronics*, vol. 54, no. 6, pp. 86–89+94, 2020.

[7] C. Bao, X. Ruan, X. Wang, D. Pan, W. Li, and K Weng, "Design of grid-connected inverters with LCL filter based on PI regulator and capacitor current feedback active damping," *Proceedings of the CSEE*, vol. 32, no. 25, pp. 133–142+19, 2012.

[8] J. Rocabert, A. Luna, F. Blaabjerg, and P. Rodríguez, "Control of power converters in AC microgrids," *IEEE Transactions on Power Electronics*, vol. 27, no. 11, pp. 4734–4749, 2012.

[9] Z. Liu, C. Zhu, Y. Wang, J. Liu, and P.. Jie, "Two improved repetitive control methods against frequency fluctuation in IPS," *Power System Technology*, vol. 042, no. 9, pp. 3014–3023, 2018.

[10] L. Zheng, F. Jiang, J. Song, Y. Gao, and M. Tian, "A discrete-time repetitive sliding mode control for voltage source inverters," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 6, no. 3, pp. 1553–1566, 2018.

[11] A. Trivedi and M. Singh, "Repetitive controller for VSIs in droop-based AC-microgrid," *IEEE Transactions on Power Electronics*, vol. 32, no. 8, pp. 6595–6604, 2017.

[12] T. Hornik and Q.-C. Zhong, "A current-control strategy for voltage-source inverters in microgrids based on H-infinity and repetitive control," *IEEE Transactions on Power Electronics*, vol. 26, no. 3, pp. 943–952, 2011.

[13] T. Hornik and Q.-C. Zhong, "Hco repetitive voltage control of grid-connected inverters with a frequency adaptive mechanism," *IET Power Electronics*, vol. 3, no. 6, pp. 925–935, 2010.

[14] R. Fei, X. C. Gong, H. S. Dao, and L. Derong, "The deadbeat control strategy of modular multilevel converter," *Proceedings of the CSEE*, vol. 37, no. 06, pp. 1753–1746, 2017.

[15] M. Odavic, V. Biagini, P. Zanchetta, M. Sumner, and M. Degano, "One-sample-period-ahead predictive current control for high-performance active shunt power filters," *IET Power Electronics*, vol. 4, no. 4, pp. 414–423, 2011.

[16] J. V. Flores, L. F. A. Pereira, G. Bonan, D. F. Coutinho, and J. M. Gomes Da Silva, "A systematic approach for robust repetitive controller design," *Control Engineering Practice*, vol. 54, no. SEP, pp. 214–222, 2016.

[17] A. Lidozzi, C. Ji, L. Solero, P. Zanchetta, and F. Crescimbini, "Digital deadbeat and repetitive combined control for a stand-alone four-leg VSI," *IEEE Transactions on Industry Applications*, vol. 53, no. 6, pp. 5624–5633, 2017.

[18] A. Trivedi and M. Singh, "Repetitive controller for VSIs in droop based AC-microgrid," *IEEE Transactions on Power Electronics*, vol. 32, no. 8, pp. 6595–6604, 2016.

Hindawi
Mathematical Problems in Engineering
Volume 2021, Article ID 3568386, 12 pages
https://doi.org/10.1155/2021/3568386

Research Article

A New Online and Offline Blended Teaching System of College English Based on Computer Internet Technology

Ping Li,[1] Hua Zhang,[1] and Sang-Bing Tsai[2]

[1]*Foreign Language School, Weifang University, Weifang, Shandong 262100, China*
[2]*Regional Green Economy Development Research Center, School of Business, Wuyi University, Nanping, China*

Correspondence should be addressed to Hua Zhang; zhanghua198316@126.com and Sang-Bing Tsai; sangbing@hotmail.com

Received 25 August 2021; Revised 17 September 2021; Accepted 4 October 2021; Published 14 December 2021

Academic Editor: Guoqiang Wang

Through the analysis and research of the multimedia teaching platform system based on computer Internet technology, this paper proposes constructing a network multimedia teaching platform system based on streaming transmission to realize the sharing of excellent teaching resources in colleges and universities and the development of education and teaching. The design and development of each module of the teaching platform are completed, and the realization technology of each module function is explained. We completed the design and implementation of some auxiliary functions of the teaching platform, including the system usage help module, resource search module, and online on-demand playback functions. After the experiment, we compare the changes of students' English anxiety before and after the experiment and conduct interviews with them. The results show that the application of the Internet-based scaffolding teaching model in college English reading teaching can alleviate students' English learning anxiety. Through interviews to understand students' feedback and evaluation of the new teaching model, it was found that most students hold positive evaluations of the Internet-based scaffolding teaching model, believing that the new teaching model can stimulate students' English learning enthusiasm and increase confidence in English learning.

1. Introduction

The Internet has also had a greater impact on college English teaching, and many teachers have adopted multimedia and other tools in their teaching. Internet teaching also brings about powerful challenges to teachers. Teachers need to have more knowledge than before. They are no longer a single source of knowledge under the traditional teaching model. Students can acquire more knowledge through the Internet, and they must also learn how to find information and how to find resources from the Internet by themselves [1, 2].

As a language course, college English is an important course at the junior high school stage. It is highly applicable and requires long-term systematic study and training. Under the traditional teaching model, teachers always feel exhausted, and students' learning enthusiasm is not optimistic. The research on the blended learning model of college English reading classrooms can cultivate the information literacy of primary school students [3, 4]. Introducing learning resources from the Internet into classroom teaching will help cultivate information awareness and literacy. Modern education has used various communication technologies and computer technologies to construct a remote teaching mode that separates teachers and students. Traditional classroom teaching has been transforming to multimedia classroom teaching and virtual classroom teaching constructed through the Internet, and more and more schools are beginning to pay attention to the construction of quality courses. Integrating the teaching resources of excellent teachers for more learners and teachers to learn from has become a problem before us, using the streaming technology of streaming media [5]. The multimedia video resources are compressed, encoded, streamed, and packaged into continuous data packets for transmission to the client. Realizing the real-time transmission and playback of excellent teaching resources provides us with a good solution [6].

The three-tier structure includes the client, WEB server, and database. After receiving the user's request, the WEB server searches the database for the resource requested by the user and then transmits it to the target through network streaming. The platform provides learners with quick query functions. Learners can not only use keywords to inquire but also inquire according to the chapters and disciplines of the knowledge they have learned. The background database of the teaching platform system stores a large amount of teaching resource information and user information. The system sets up a unified administrator to manage the teaching resource information and users uniformly to ensure the safety and reliability of courseware resources and user information. After the user logs in to the platform, he queries and selects the resource information he needs and realizes the on-demand viewing function through the streaming media server and streaming video player. The Internet-based scaffolding teaching model integrates the Internet teaching concept to better realize the classroom teaching effect.

2. Related Work

The United States proposes building a "national information infrastructure" [7]. With the help of personal computers and the Internet, the education industry has realized the digitization of education environment, education content, and education management. For example, the United States initiated the "Digital Library Initiative Project" and "National Digital Library Project" to popularize digital libraries in cities and schools; the digital publishing industry has sprung up, with the help of Internet distribution channels, and the media, using online payment as the main means, changed the pattern of the traditional publishing industry [8].

The integration of computer networks and language teaching does not rely on simple research in one or two disciplines to solve problems. Its emergence is based on the research of multiple theories, and it is the continuous optimization and integration of current discipline research [9]. The rapid development of computer network technology and alternate learning theories are integrated with each other, which regulates the development characteristics of language teaching in this environment. Behaviorist computer-aided language teaching, interactive computer-aided language teaching, and integrated computer-aided language teaching have appeared [10]. While fully affirming the guiding significance of learning theory to teaching practice, researchers have carried out more in-depth research [11]. Relevance learning theory is the best choice for the construction of language teaching models in the current computer network environment [12]. It believes that teaching is the connection between information exchange and the process of circulation, and information technology is the only prerequisite medium. This theory is more suitable for cultivating students to realize individualized and creative independent learning. Researchers generally suggest that the various learning theories such as behaviorism, cognitivism, constructivism, and relevance should be scientifically connected and integrated [13].

The first category is foreign commercial platforms, which generally cost more to develop and use, but they can provide users with good services and have better platform performance. These platforms have strong technical force and are mature and stable but generally have higher service costs, such as the Blackboard teaching platform. The second category is domestic commercial platforms, such as the THOEL online teaching platform of Tsinghua University and the sky classroom developed by Nanjing Yixue Education Software Co., Ltd. The domestic online teaching platform is more suitable for China's teaching needs and is more in line with the reality of China's higher education. The third category is a network teaching platform independently developed by universities according to the needs of their teachers, students, and teaching and is only open and used within the scope of the university. This type of teaching platform is developed according to the characteristics and needs of the university itself, and the manpower and material resources consumed by the development are not high but may be inferior to the first two types of platforms in terms of compatibility and adaptability. The fourth category is open-source software. Since these software programs have open source codes, users can partially modify the source code of this type of software to meet their own teaching or learning needs [14, 15]. There are many such platforms, such as the Moodle platform.

Related scholars put forward the concept of blended learning to provide students with personalized help, find and formulate learning resources and methods in line with their personality and learning style, and organically integrate these personalized resources to provide targeted development for students [16]. Researchers emphasize that blended learning is an optimized learning model that integrates modern educational technologies that can assist in achieving expected learning goals, while taking projects as the core [17]. Relevant scholars have proposed that blended learning is an innovative education form centered on learners, which integrates various learning theories, diversified teaching methods, teaching concepts, and advanced educational technology [18–20].

Academic self-efficacy directly affects the degree of participation in online open courses and indirectly affects the persistence of online open courses. The existence of teachers can enhance learners' participation and directly affect learners' learning persistence and improve their desire to complete the course. The perceived usefulness of learners when using the online open course platform greatly affects their learning participation, but it does not directly affect the persistence of online open course learning. The ease of use of the online open course platform does not significantly affect learner participation, but it directly affects the persistence of online open course learning. Participation in online open courses directly affects the persistence of online open courses. Therefore, when designing online open courses, teachers should formulate specific plans to meet the needs of learners and encourage them to actively participate in online open courses, so as to improve course completion [21–23].

3. College Students' English Mixed Teaching Mode Supported by Computer Internet Technology

3.1. Key Technologies to Support Computer Internet Mobile Learning. The technologies that support computer Internet mobile learning include mobile terminal equipment and wireless network access methods. Computer Internet mobile learning terminal equipment is one of the three major elements of the computer Internet mobile learning system environment. It is a basic part of the development of computer Internet mobile learning and a necessary condition for the implementation of computer Internet mobile learning activities. The online and offline hybrid teaching mode of college English using computer Internet technology is shown in Figure 1.

3.2. Analysis of Blended Teaching

3.2.1. A Mixture of Interpersonal and Human-Computer Interaction in and out of Class. The interpersonal interaction of traditional classroom teaching takes place in class. This kind of interaction is a one-way interaction. It supports the interaction between teachers and students in class through modern multimedia technology devices such as mobile smartphones, computers, and projectors through the implementation of classroom barrage, classroom real-time rush answers, lottery questions, and random group discussions in the classroom. After class, through resource sharing, group collaboration, teacher-student Q&A, student-student discussion, and so forth, this multidirectional interaction is the interaction between teachers, students, and teaching media. It is also in class, offline, and online.

3.2.2. Mixed Teaching Media Supported by Computer Internet Technology. The mixed teaching supported by computer Internet technology can choose from various teaching media with rich functions. In the organization of teaching, in addition to traditional teaching media "books, blackboards, projectors," and so forth, modern mobile communication technology terminal equipment, such as "mobile phones, tablets, and notebooks," can be used as interactive tools in the classroom and can also be used outside the classroom.

3.2.3. Mixed Teaching Environment Supported by Computer Internet Technology. Blended teaching is not a fixed, simple mixture of "offline teaching from person to person" and "online self-directed learning." The hybrid teaching environment enriches traditional classroom teaching methods, uses mobile devices to communicate and interact online, and mobilizes students' interest in learning and classroom atmosphere. Mobile devices have the characteristics of wireless transmission. Teachers and students can use video and voice calls, electronic whiteboards, and other functions outside the classroom to perform wireless projection and multiscreen interaction, so as to realize online teaching.

3.2.4. Mixing of Teaching Content and Resources. College English requires a hybrid teaching model supported by computer and Internet technology. In addition to the traditional textbook knowledge, its teaching content also includes electronic courseware, video, audio, electronic documents, and other learning materials uploaded by the teacher. The content of the learning materials can be passed by the teacher. Students can use mobile devices to obtain teaching resources in class or at any other time, supplemented by classroom learning, or perform informal contextual learning. In informal situations, students can search for corresponding learning resources through WEB, foreign language learning APP, WeChat official account, and so forth according to their individual learning needs.

3.2.5. "Dominant-Subject" Teaching Structure Is Mixed. The constructivist teaching concept believes that teaching should take learners' original knowledge and experience as the growth point of new knowledge and guide learners to actively construct new knowledge and experience from their original knowledge and experience. Teachers and students as well as students and students should communicate and question one another in the process of exploring certain issues together. In this exploration and interaction, it is necessary to give full play to the guiding role of teachers and fully reflect the main position of students. Whether it is offline collective face-to-face instruction or online individual instruction, teachers are the planners, organizers, and guides of the entire teaching process. The appropriate teaching strategy should be selected according to the student's learning situation. Through meaning construction and repeated training, students are guided from understanding to mastering the structure of words, grammar, and sentence structure, until they are used proficiently, and the language knowledge is transformed into language skills.

3.3. Data Mining of College Students' English Blended Teaching

3.3.1. ID3 Algorithm. S_i is the number of samples in class C_i. The class label has m different values, and m different classes C_i ($i = 1, 2, \ldots, m$) are defined.

$$I(S_0, S_1, S_2, \ldots, S_{m-1}) = \prod_{i=0}^{m-1} \log_2 |P_i|. \tag{1}$$

In the above formula, P_i is the probability of any sample belonging to C_i. Since the information is expressed as a binary code, it is based on the logarithmic function 2.

Suppose that attribute A has v different values {a1. a2 ... av}. The information entropy divided into subsets by A is given by

$$E(A) = \text{Info}\left(\frac{D_j}{(D)}\right) \cdot \prod_{j=0}^{v-1} |D_j|,$$

$$D_j = 1 - S_{1j} - S_{2j} - \cdots - S_{mj}, \tag{2}$$

$$I\left(S_{0j} - S_{1j} - S_{2j}, \ldots, S_{(m-1)j}\right) = P_{ij} \cdot \prod_{i=0}^{m-1} \log_2 |P_{ij}|.$$

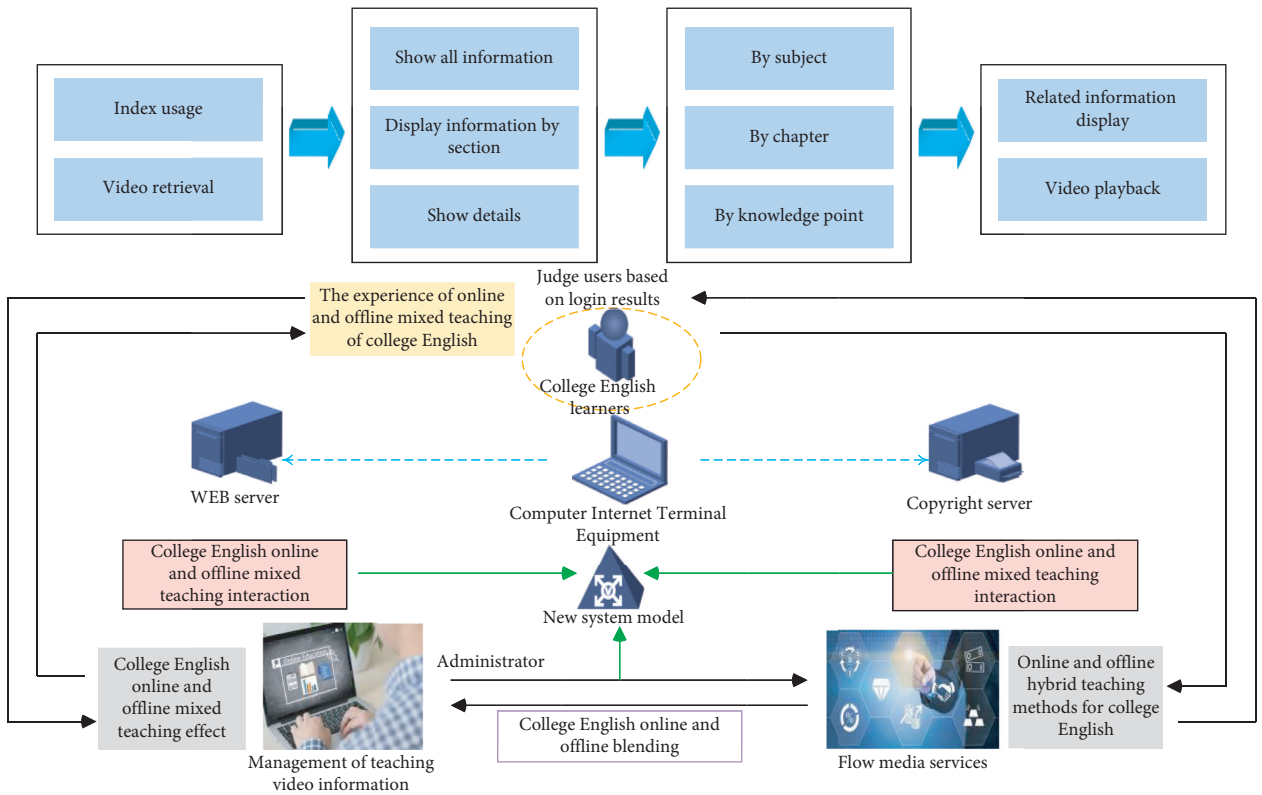


Figure 1: Online and offline mixed teaching mode of college English using computer Internet technology.

In the above formula, P_{ij} is the probability that the sample in S_j belongs to class C_i. The information gain obtained by branching on attribute A can be obtained by the following formula:

$$Z(A) = E(A) + I(S_0 - S_1 - S_2 - \cdots - S_{m-1}). \quad (3)$$

3.3.2. C4.5 Algorithm. When the ID3 algorithm splits, it pays more attention to splitting multivalued attributes. Such a division looks very fine, but it is actually meaningless. But ID3's improved algorithm C4.5 makes up for this shortcoming, because the C4.5 algorithm introduces the concept of gain rate to split the attributes.

The C4.5 algorithm uses a formula to calculate the split information; the formula is

$$\gamma(D) = \frac{1}{D} \cdot \log_2 D_j \cdot \prod_{j=0}^{v-1} |D_j|. \quad (4)$$

Then we select the attribute with the largest gain rate for division. The calculation method is similar to ID3. The formula is

$$\chi(A) = \frac{Z(A)}{\gamma(A)}. \quad (5)$$

4. System Analysis and Design of Online and Offline Mixed Teaching Platform

4.1. The Overall Design of College English Online and Offline Mixed Teaching Platform. The video teaching information contained in the platform system is prepared in advance or collected in real time by video and audio acquisition equipment, and then the prepared streaming media format files are released through the streaming media server, and the web page information is released through the WEB server.

After the user enters, the platform system first presents the login page to the user, asking the user to enter their user name and password. The platform system will enter the background database for query and comparison. The identity is a system administrator or a learner. Users can browse the resource information of the platform system according to their needs, select the learning resources they need, and conduct online learning. If it is the first time to register, you can also view the instructions for use of the system; if the user is an administrator, you will enter the administrator interface. The administrator can view current user information and video resource information. If the user information is not found in the background database, the user will be prompted whether the user name or password has been entered incorrectly. The system workflow of this platform is shown in Figure 2.

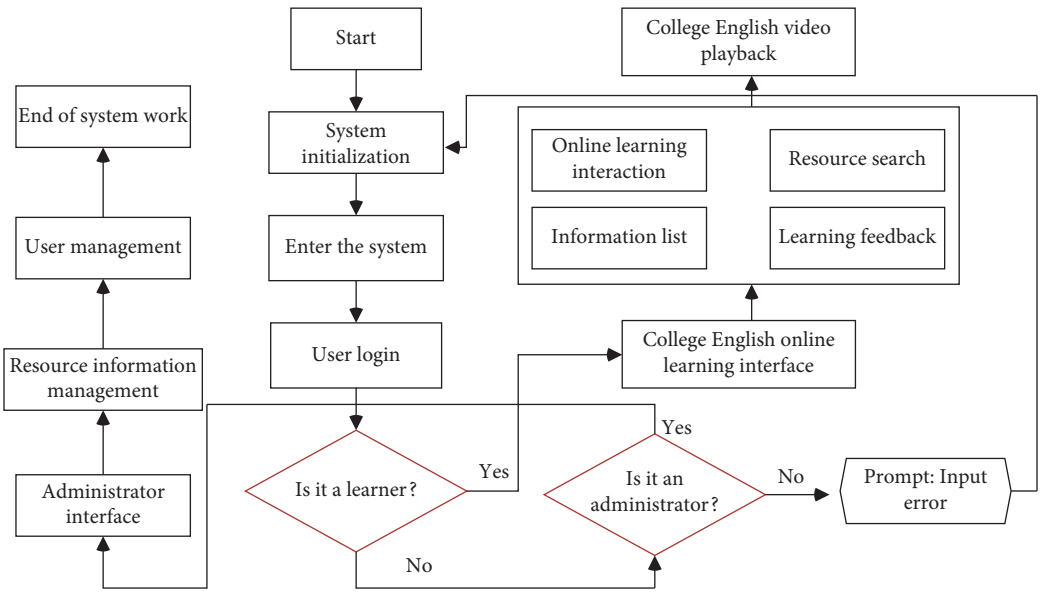


Figure 2: Workflow of college English online and offline mixed teaching system.

4.2. Production and Release of Streaming Media Courseware. Multimedia courseware is the use of multimedia technology to transform boring course content into vivid multimedia information for learners to learn. The use of multimedia technology can express the knowledge and concepts that learners find difficult, abstract, and difficult to understand in the form of animation, which is convenient to learners' learning and understanding. The production of multimedia courseware first needs to convert teaching materials into digital information, generate a digital video file, and then stream the file to make a stream format file and then add the logo information we need to the stream format file. In the technology, parameters such as Marker and Command can be added to the stream file. This identification information will make the client software produce some effects we need when playing the stream file. When the client software reads the label, it will generate a Marker Hit event and pass a parameter. This parameter is the content corresponding to the label.

At the same time, the content of the label will be displayed in the text box of the client. This shows that we should write the stream format file. Some labels are added to the corresponding positions, so that the entire online courseware can be coordinated. In order to enable the client to use the browser to watch the courseware, we must also use Dream Weaver software to make the corresponding web page, insert the Media Player plug-in in the web page, and use the VBscript scripting language to write the corresponding processing program to handle the Marker Hit event. You can also add some control buttons to realize the user's playback control of the video information. The logical structure of streaming media file playback and information flow delivery is shown in Figure 3.

For the production of streaming media courseware, you can choose the teacher's Windows Media Encoder tool to record the teacher's real-time lectures into ASF format files, and you can also use the Windows Media Author tool to convert the recorded audiovisual teaching resource files into ASF streaming format for release. Both methods require Media ASF Indexer to add tags and descriptions to the stream format file to achieve the synchronization display of the relevant learning materials.

4.2.1. Publishing Streaming Media Courseware Based on WEB. Publishing streaming media courseware based on WEB is to put the prepared streaming media courseware on the WEB server, and users can download and watch it through the browser or watch it online. The efficiency of publishing courseware in this way is relatively low, and there is no real-time guarantee, and it cannot support too many users to use it at the same time.

4.2.2. Release Based on Streaming Media Server. This method of publishing streaming media courseware should be realized by using a streaming media server. Windows Media Server is a component provided by the Windows 2010 server operating system, which can stream multimedia content from low-bandwidth to high-bandwidth networks. The streaming media server component is composed of Windows Media Component Service and Windows Media Manager. Windows Media Component service is a series of services running on Microsoft Windows 2010 server. These services can transmit audiovisual information to users through unicast, multicast, and broadcast. Windows Media

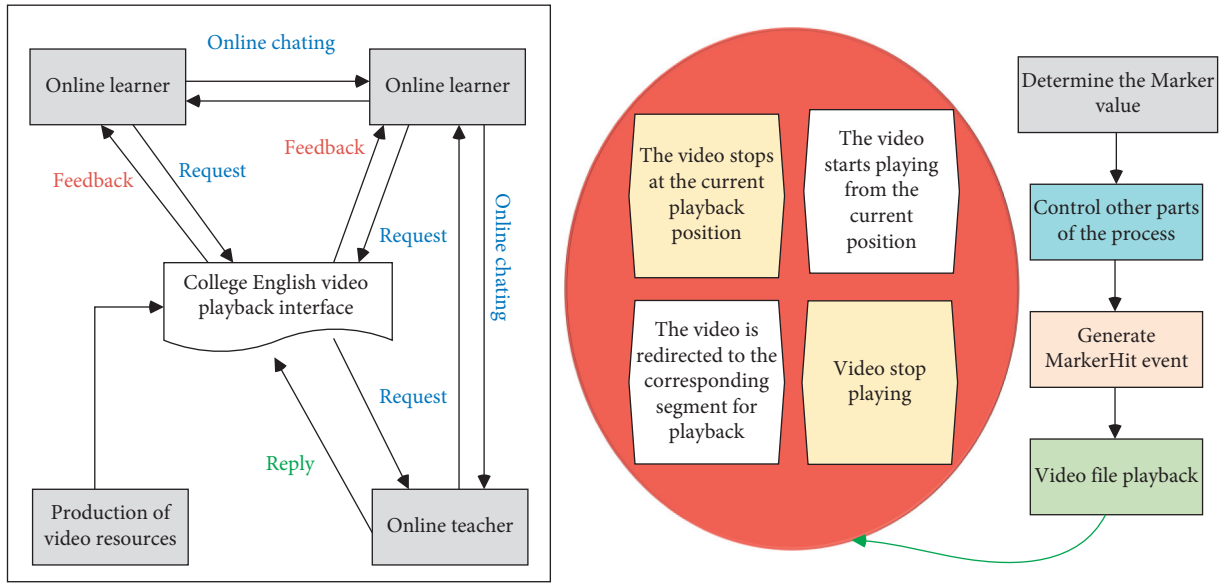


Figure 3: Logic diagram of streaming media playback and information flow delivery.

Manager is a series of WEB pages used to manage Windows Media Component services, which can manage and control local servers.

4.3. The Design of Various Functional Modules of College English Online and Offline Mixed Teaching Platform

4.3.1. Design of User Login Module. The system of this platform must be registered as a member to log in to the system for learning. The user must enter his user name and password through the login interface before entering the system of this platform for learning. After the system obtains the account and password entered by the user, it queries the USER data table for a record that matches the account password through a query statement. If rs is empty, it means that the user's information does not exist in the database; that is, the user is not a registered user of the platform system or the user name or password is entered incorrectly, and the user is prompted to log in again. If rs is not empty, it means that the user is a normal registered user; that is, the login is successful, and then the user's identity is judged according to the user's type. User type information is an important identification information for users of the system on this platform. Different identification information means different user identities and also means that they have different operating permissions. Therefore, the identification information will always exist until the user logs out of the system.

First we use the request object to obtain the user name and password entered by the user to ensure that they are not empty and then create an instance of the recordset object rs, find the record consistent with the information entered by the user from the user table USER, and assign it to rs. If it is not empty, we save the value type of the user type in the variable usertype. After obtaining the value of the usertype variable, the user's identity information can be determined, where 1 corresponds to the administrator, 2 corresponds to the ordinary learner, and the web page will jump to the corresponding page. If the value of usertype is neither 1 nor 2, it means that the user entered the user name or password incorrectly, and the user still stays on the original page, waiting for the user to confirm the identity, and jump to the corresponding interface. In order to prevent unnecessary losses caused by malicious access to the platform system, the user's cookies value is set when the user logs in. This value will record the user's login information and save it until the user logs out of the system. Each time a user visits a page, the user will first check whether the user's cookies value exists.

4.3.2. The Design of the Administrator Module. The user management flowchart is shown in Figure 4. The video playback interface is an important module of the platform system, and its function is to realize streaming media playback. This interface can integrate real-time teaching videos of excellent teachers for users to watch on demand. At the same time, teachers' teaching plans will be displayed synchronously on key and difficult knowledge points to allow students to deepen their understanding of this part of knowledge. The combination of this interface and the server can also provide online users with real-time and non-real-time online communication.

5. Research Results and Analysis

5.1. Analysis of the Results of Students' College English Pretest. In order to ensure that the reading anxiety levels of the two classes before the experiment are the same, before the experiment, this paper analyzes the results of the questionnaires of the two classes from three dimensions (text comprehension anxiety, emotional anxiety, and cultural background anxiety).

It can be seen from Figure 5 that, before the experiment, from the perspective of text comprehension anxiety, the

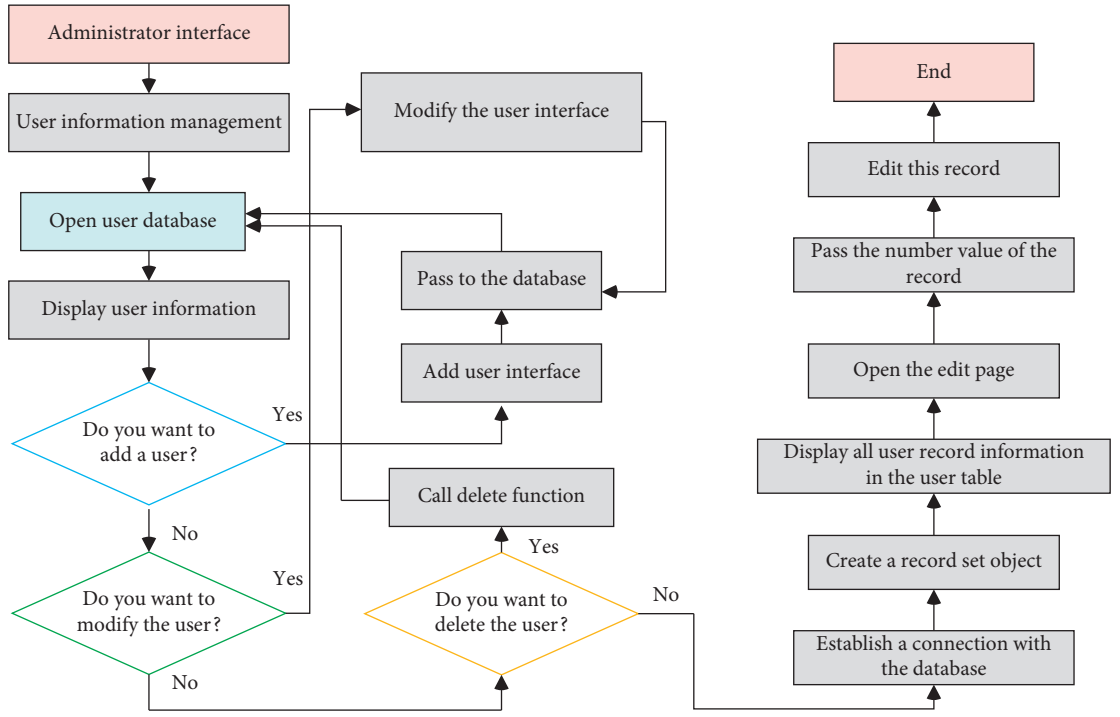


Figure 4: Flowchart of user management for online and offline mixed teaching of college English.

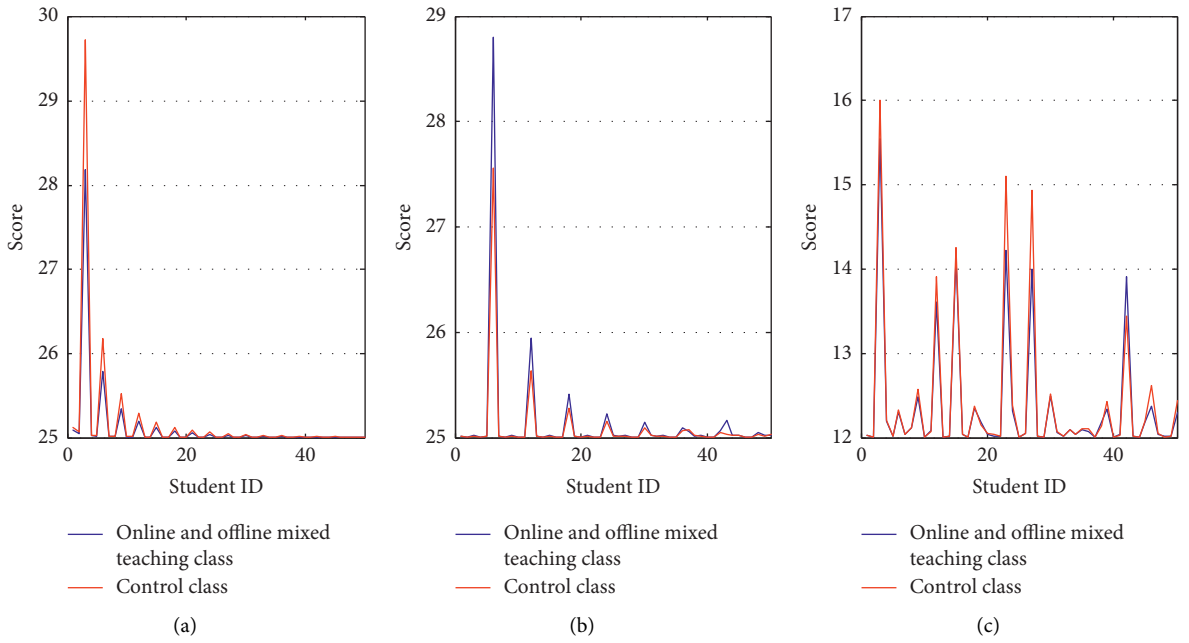


Figure 5: Pretest statistics of three dimensions of college English learning in online and offline mixed teaching classes and control classes. (a) Text understanding anxiety. (b) Emotional anxiety. (c) Cultural background anxiety.

average value of the control class is about 25.8 points, and the average value of the online and offline mixed teaching class is about 25.6 points. It can be seen that the two classes are in terms of text comprehension anxiety. From the perspective of emotional anxiety, the average value of the control class is about 25.4 points, and the average value of the online and offline mixed teaching class is about 25.1 points, indicating that the average gap in emotional anxiety is also small. From the perspective of cultural background anxiety, the average value of the control class is about 13.1 points, and the average value of the online and offline mixed teaching class is about 13.3 points. It can be seen that the two classes have a small difference in the scores of cultural background anxiety. Before the experiment, the students in

both classes had the same degree of reading anxiety during their English reading learning process, and the two classes met the experimental conditions.

5.2. Analysis of Posttest Results of Students Learning College English. It can be seen from Figure 6 that, after the teaching experiment, the average posttest reading anxiety of the online and offline mixed teaching class is about 51.7 points, and the average of the control class is about 65.2 points. The average posttest learning anxiety of the online and offline mixed teaching classes was lower than the average of the control classes. These data show that, after teaching experiments, students in the online and offline mixed teaching classes have a certain degree of relief in English reading anxiety, while the students in the control class may have higher learning anxiety due to the final exam and other reasons.

In this paper, the posttest results of the questionnaires of the two classes are also descriptive statistics from three dimensions (text comprehension anxiety, emotional anxiety, and cultural background anxiety), as shown in Figure 7.

It can be seen from Figure 7 that, after the experiment, from the perspective of text comprehension anxiety, the average value of the control class is about 27.5 points, and the average value of the online and offline mixed teaching class is about 25.4 points. There are differences in anxiety; from the perspective of emotional anxiety, the average value of the control class is about 28.6 points, and the average value of the online and offline mixed teaching class is about 25.9 points, indicating that there is a large gap in the average value of emotional anxiety; from the perspective of cultural background anxiety, the average value of the control class is about 13.8 points, and the average value of the online and offline mixed teaching class is about 12.9 points. It can be seen that the scores of the two classes in the cultural background anxiety dimension are slightly different. In summary, the two classes have some gaps in their college English scores after the experiment.

The learning of cultural background knowledge requires students to accumulate for a long time, so, in a semester, students will still feel a lack of background knowledge of articles. Therefore, through the analysis and discussion of the above data, we can conclude that the English reading anxiety level of the two classes of students has significant differences in the two dimensions of text comprehension anxiety and emotional anxiety. Although there is no significant difference in cultural background anxiety, there is no significant difference. We can see that there is still a slight gap in the scores of the two classes in this dimension after the experiment.

Because this article compares and analyzes the English reading anxiety levels of the two classes before the experiment, it is concluded that they have the same teaching content, teaching teachers, and class schedules. From this, we can verify the first research question; that is, the Internet-based scaffolding teaching model can increase reading confidence, reduce students' English reading anxiety, and gain a sense of learning achievement.

5.3. Analysis of Pre- and Posttest Results of Students' Academic Performance in Online and Offline Mixed Teaching Classes and Control Classes. This article conducts a horizontal analysis of the data in the first two parts. In order to analyze the data more accurately, this article will make descriptive statistics on the English learning performance values of the two classes of students to make the experimental data more convincing.

Before the experiment, the academic performance of the online and offline mixed teaching class was similar to that of the control class. It can be seen from Figure 8 that, after the experiment, the average score of the online and offline mixed teaching class has increased by about 8.3 points. It can be seen that, after the experiment, the English learning performance value of the students in the online and offline mixed teaching class has greatly improved. According to Satio's standard of academic performance, 20–46 is divided into low performance level, 47–68 is divided into medium performance level, and 69–100 is divided into high performance level, which shows the learning of students in online and offline mixed teaching classes. A high level of achievement has been reached. This shows that the new teaching model has a significant effect on improving students' English learning performance. However, compared with online and offline mixed teaching classes, the average academic performance of the control class before and after the experiment is lower. This shows that, after a semester of teaching, the English learning performance of the students in the control class has not improved and is still at a low level.

In order to further verify the practical effect of the new teaching model, this article will conduct specific analysis and discussion from three dimensions: text comprehension score, learning emotion score, and cultural background score.

5.3.1. Text Comprehension Score. The data in Figure 9 show the postdescriptive statistical results of text comprehension scores in online and offline mixed teaching classes and control classes. From the data in the figure, it can be seen that the average text comprehension score of the online and offline mixed teaching class after the experiment is about 84.5 points, while the average value of the control class is about 77.5 points. It can be seen that, from the perspective of the text comprehension score, after the experiment, the text comprehension score of the online and offline hybrid teaching class is better than that of the control class.

5.3.2. Learning Emotion Score. Figure 10 shows the descriptive statistical results. It can be seen that, from the perspective of emotional performance, after the experiment, the emotional score of the online and offline mixed teaching class has increased rapidly.

5.3.3. Cultural Background Score. Figure 11 shows the descriptive statistical results of the posttest of cultural background scores in online and offline mixed teaching classes

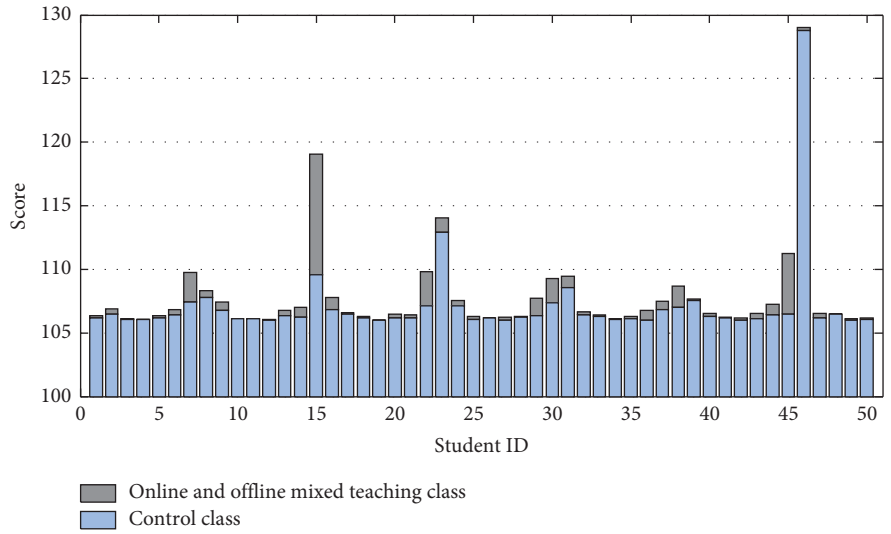


Figure 6: Comparison results of posttest description of learning anxiety of students in control classes and online and offline mixed teaching classes.

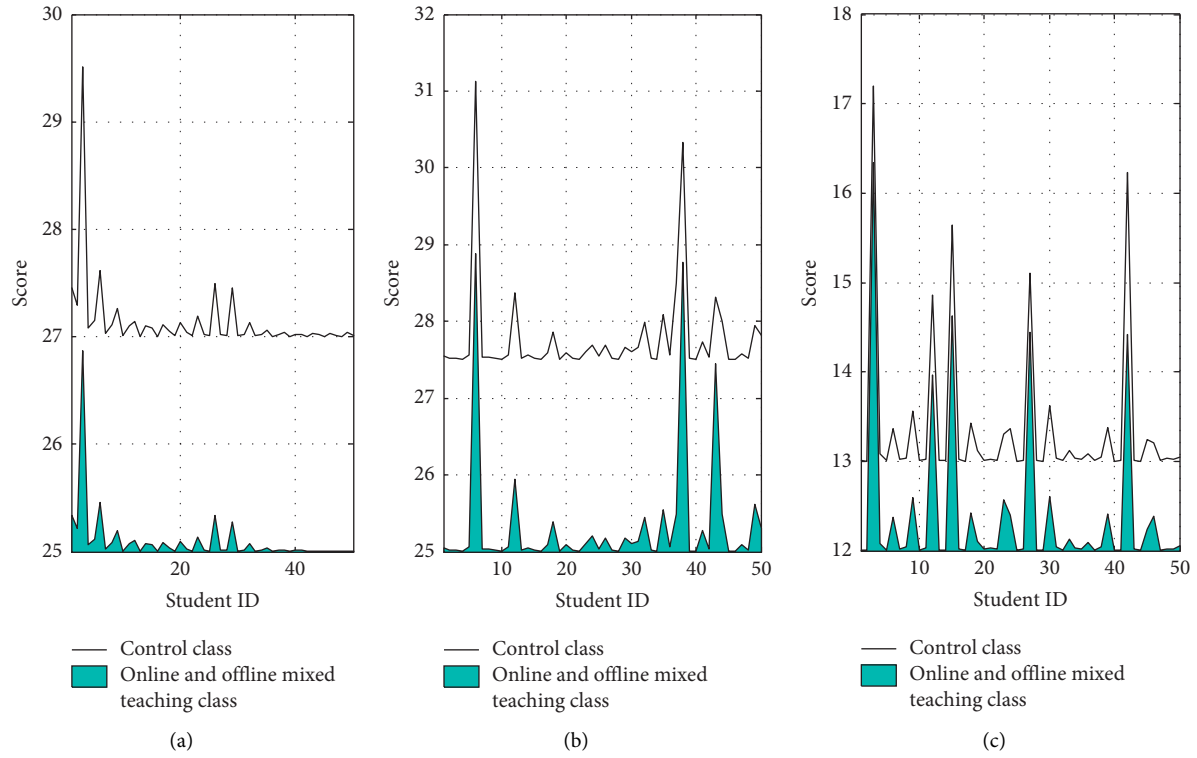


Figure 7: Descriptive statistics of the three dimensions of college English learning in online and offline mixed teaching classes and control classes. (a) Text comprehension anxiety posttest. (b) Emotional anxiety posttest. (c) Cultural background anxiety posttest.

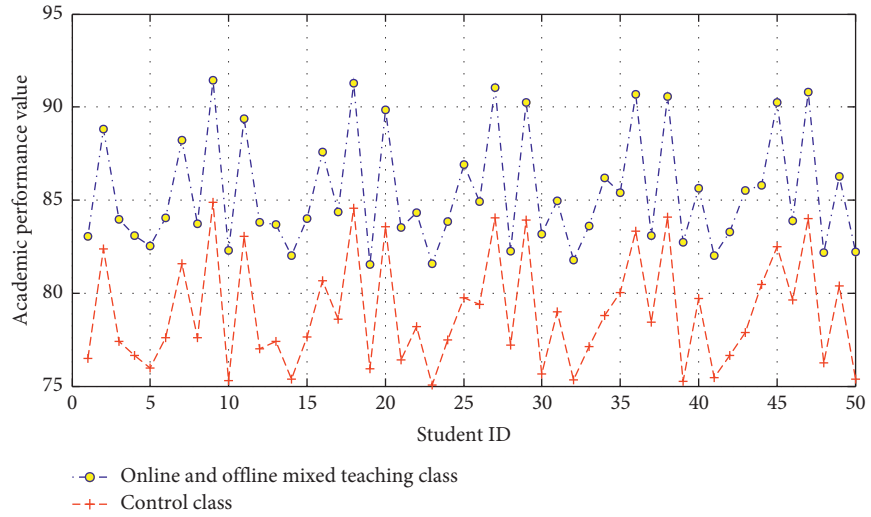


Figure 8: Posttest descriptive statistics of learning performance in online and offline mixed teaching classes and control classes.

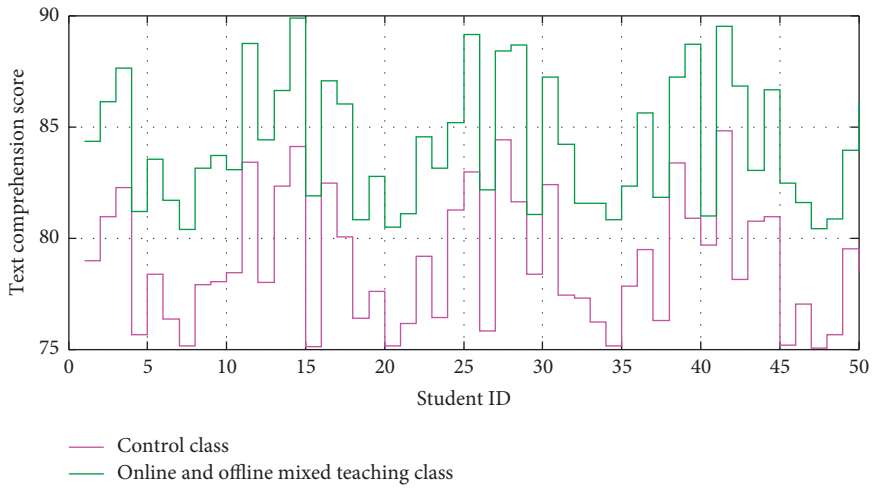


Figure 9: Posttest descriptive statistics of text comprehension scores in online and offline mixed teaching classes and control classes.

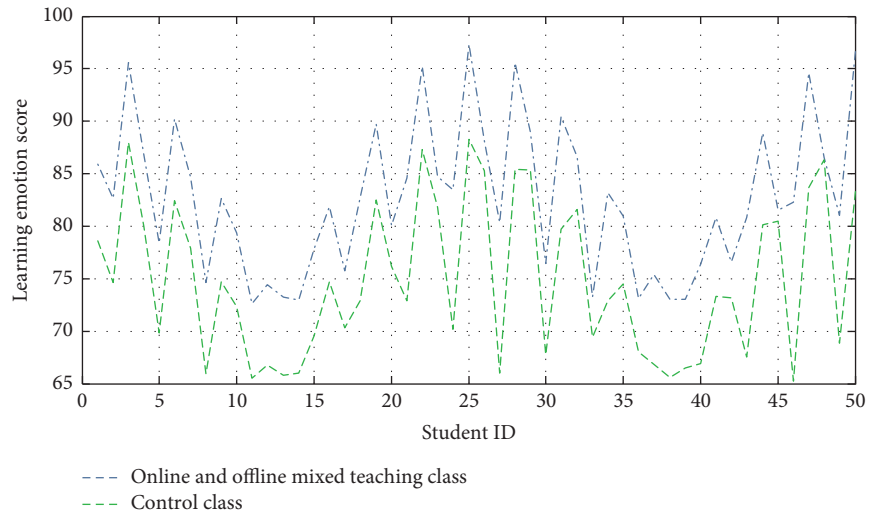


Figure 10: Posttest descriptive statistics of learning emotion scores in online and offline mixed teaching classes and control classes.

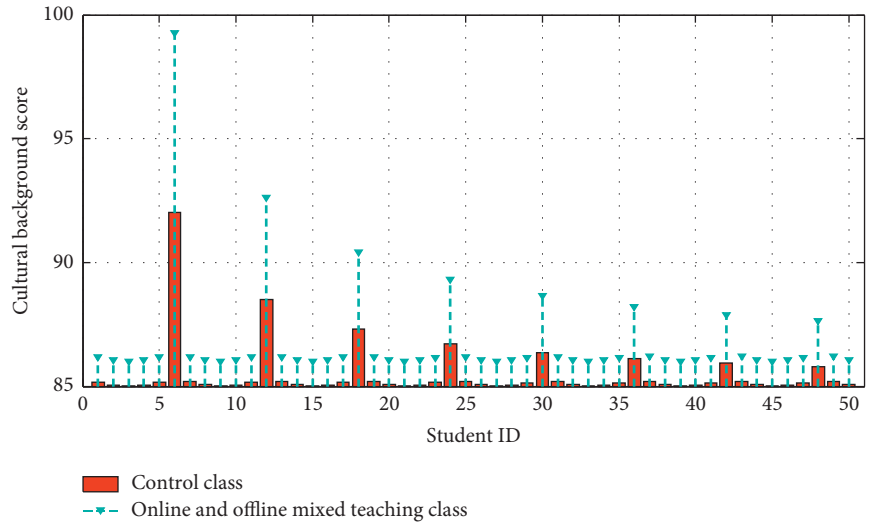


Figure 11: Posttest descriptive statistics of cultural background scores in online and offline mixed teaching classes and control classes.

and control classes. It can be seen that, after the experiment, the average value of the cultural background score of the online and offline mixed teaching class is about 87.9 points, and the average value of the control class is about 86.1 points.

6. Conclusion

This article expounds the current domestic and foreign development of computer Internet technology and introduces the application of computer Internet technology in education and teaching. This paper explains the principle of computer Internet technology and its transmission protocol and emphatically introduces the streaming transmission and on-demand playback methods suitable for the platform system. Media selection theory, learning theory, and individualized learning theory all provide a good theoretical foundation for the application of computer Internet technology to college teaching. The system requirements analysis and design process of the platform are described, and the function realization of each module is described. The Internet-based scaffolding teaching model can improve students' English reading performance. Comparing the scores before and after reading the test papers, it is found that the scores of the students in both classes have improved, but the reading scores of the online and offline mixed teaching classes have improved even higher; and, through interviews with students, this article found that students in online and offline hybrid teaching classes can actively participate in learning activities and complete learning tasks assigned by the teacher. The only difference is the teaching method. This shows that the Internet-based scaffolding teaching model helps students develop a sense of achievement and self-confidence in reading and develop a sense of autonomous learning. Students' attitudes towards the Internet-based scaffolding teaching model are positive. Through interviews with students, this article learned that students believe that the Internet-based scaffolding teaching model can give them sufficient time for autonomous learning before class.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] Y. Yao, "Blended teaching reform of higher vocational education based on addie teaching design model," *Journal of Frontiers of Society, Science and Technology*, vol. 1, no. 5, pp. 60–65, 2021.

[2] Z. Liu, "Research on college english mixed teaching mode under information technology environment," *Revista Argentina de Clinica Psicologica*, vol. 29, no. 5, pp. 1673–1681, 2020.

[3] H. Sun, "A SPOC teaching mode of college english translation based on "rain classroom"," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 14, no. 17, pp. 182–193, 2019.

[4] Y. Li, "Construction of mixed classroom in higher vocational english teaching under MOOC," *Online Learning*, vol. 19, pp. 15–65, 2019.

[5] Y. Wang and X. Ma, "Research on blended teaching ability of college english teachers," *Open Journal of Social Sciences*, vol. 8, no. 12, pp. 308–319, 2021.

[6] Y. Shisheng, "Study on the mixed teaching model of college teaching under the background of educational informationization," *Studies in Literature and Language*, vol. 20, no. 3, pp. 28–32, 2020.

[7] Z. Zhou, "On the lesson design of online college english class during the COVID-19 pandemic," *Theory and Practice in Language Studies*, vol. 10, no. 11, pp. 1484–1488, 2020.

[8] L. Yang, "The teaching design of computer network's flipped classroom based on fanya SPOC teaching platform," *Sino-US English Teaching*, vol. 15, no. 2, pp. 87–91, 2018.

[9] Y. Gao, "Blended teaching strategies for art design major courses in colleges," *International Journal of Emerging*

Technologies in Learning (iJET), vol. 15, no. 24, pp. 145–158, 2020.

[10] Z. Xu and Y. Shi, "Application of constructivist theory in flipped classroom—take college english teaching as a case study," *Theory and Practice in Language Studies*, vol. 8, no. 7, pp. 880–887, 2018.

[11] Y. Jia and L. Zhang, "Research and application of online SPOC teaching mode in analog circuit course," *International Journal of Educational Technology in Higher Education*, vol. 18, no. 1, pp. 1–14, 2021.

[12] S. Hua and F. Liu, "A new hybrid teaching model for a psychology course," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 16, no. 3, pp. 206–219, 2021.

[13] C. Wang, "Employing blended learning to enhance learners' english conversation: a preliminary study of teaching with hitutor," *Education and Information Technologies*, vol. 26, no. 2, pp. 2407–2425, 2021.

[14] I. Mutambik, "The Role of e-learning in studying english as a foreign language in Saudi Arabia: students' and teachers' perspectives," *English Language Teaching*, vol. 11, no. 5, pp. 74–83, 2018.

[15] M. Huang, Y. Shi, and X. Yang, "Emergency remote teaching of english as a foreign language during COVID-19: perspectives from a university in China," *IJERI: International Journal of Educational Research and Innovation*, vol. 1, no. 15, pp. 400–418, 2021.

[16] X. U. Li, "English and American literature teaching in colleges and universities in the new media age," *Higher Education of Social Science*, vol. 18, no. 1, pp. 41–44, 2020.

[17] S. R. Milonm and H. Hasanand Ipban, "Students' perception towards technology in learning english as a foreign language: a case study of higher secondary students of Pabna, Bangladesh," *IOSR Journal of Humanities and Social Science*, vol. 22, no. 6, pp. 47–53, 2017.

[18] M. Li, Z. An, and M. Ren, "Study on student-centered artificial intelligence online teaching+ home learning model during the COVID-19 epidemic," *Inteligencia Artificial*, vol. 23, no. 66, pp. 51–65, 2020.

[19] N. Aulia and R. W. Batubara, "Teachers' perception towards online english learning using distance education system at rural schools area, Thailand," *International Journal of Multicultural and Multireligious Understanding*, vol. 8, no. 1, pp. 338–347, 2021.

[20] J. Irudayasamy, S. Y. Uba, and C. A. Hankins, "Exploration and exploitation of mobile apps for english language teaching: a critical review," *English Language Teaching*, vol. 14, no. 4, pp. 43–54, 2021.

[21] T. Grubljesic, P. S. Coelho, and J. Jaklic, "The shift to socio-organizational drivers of business intelligence and analytics acceptance," *Journal of Organizational and End User Computing*, vol. 31, no. 2, pp. 37–64, 2019.

[22] L. Z. Zhang, M. Mouritsen, and J. R. Miller, "Role of perceived value in acceptance of "bring your own device policy"," *Journal of Organizational and End User Computing*, vol. 31, no. 2, pp. 65–82, 2019.

[23] A. Shahri, M. Hosseini, K. Phalp, J. Taylor, and R. Ali, "How to engineer gamification: the consensus, the best practice and the grey areas," *Journal of Organizational and End User Computing*, vol. 31, no. 1, pp. 39–60, 2019.

Hindawi
Mathematical Problems in Engineering
Volume 2021, Article ID 1401802, 12 pages
https://doi.org/10.1155/2021/1401802

Research Article

Reinforcement Learning-Based Multiple Constraint Electric Vehicle Charging Service Scheduling

Yongguang Liu,[1] Wei Chen ,[2] and Zhu Huang[2]

[1]*Henan Xj Metering Co., Ltd, Xuchang 450061, Henan, China*
[2]*Hefei University of Technology, Hefei City 230041, Anhui Province, China*

Correspondence should be addressed to Wei Chen; 2020110419@mail.hfut.edu.cn

Received 24 August 2021; Accepted 21 October 2021; Published 15 November 2021

Academic Editor: Mohamed El Ghami

The popularization of electric vehicles faces problems such as difficulty in charging, difficulty in selecting fast charging locations, and comprehensive consideration of multiple factors and vehicle interactions. With the increasingly mature application of navigation technology in vehicle-road coordination and other aspects, the proposal of an optimal dynamic charging method for electric fleets based on adaptive learning makes it possible for edge computing to process electric fleets to effectively execute the optimal route charging plan. We propose a method of electric vehicle charging service scheduling based on reinforcement learning. First, an intelligent transportation system is proposed, and on this basis a framework for the interaction between fast charging stations and electric vehicles is established. Subsequently, a dynamic travel time model for traffic sections was established. Based on the habits of electric vehicle owners, an electric vehicle charging navigation model and a reinforcement learning reward model were proposed. Finally, an electric vehicle charging navigation scheduling method is proposed to optimize the service resources of the fast charging stations in the area. The simulation results show that the method balances the charging load between stations, can effectively improve the charging efficiency of electric vehicles, and increases user satisfaction.

1. Introduction

With the extensive development of electric vehicles in various countries around the world, the number of electric vehicles is increasing, and problems such as difficulty in charging electric vehicles, serious line losses, voltage drops, charging safety, and severe peaks are expected [1–3]. Electric vehicle charging and charging path planning should receive more attention. For electric vehicles whose driving time is longer than the nondriving time, fast charging is an important power supplement method [4, 5]. The disorderly charging of electric vehicles would not only cause congestion of fast charging stations, which increases the burden on the regional grid, but also result in concentrated charging times causing problems such as transformer overload and increased peak-to-valley difference, which is not conducive to the safe operation of the distribution network [6, 7]. Therefore, reasonable guidance and charging scheduling for vehicles with fast charging needs are beneficial to the alleviation of the burden on the regional grid while meeting the charging needs [8, 9].

In response to the above problems, scholars at home and abroad have conducted some research. In [10], we studied the uniform charging node in [11] and extended it to the nonuniform charging node in [12] by solving the mixed integer nonlinear programming problem (MINLP) of the single vehicle. The remaining energy of the vehicle on each node is expressed as a dynamic programming (DP) problem for a single electric vehicle path problem, and a DP-based algorithm is provided to determine the optimal path and charging strategy of the electric vehicle subflow level. In [13], we proposed a distributed electric vehicle path selection system based on the distributed ant colony algorithm (ACA). The distributed architecture minimizes the total travel of electric vehicles to the destination by proposing a set of nearest fast charging stations. In [14], we proposed an improved Dijkstra method to solve the multiobjective optimization problem and obtained a multiobjective

optimization function including travel time, fast charging station number of vehicles, and charging load, thereby optimizing electric vehicle charging path planning and alleviating fast charging stations. The lack of surrounding traffic congestion reduces waiting time and improves the availability of charging facilities.

The above literature has its own characteristics regarding charging route navigation and charging scheduling, but when studying electric vehicle charging route navigation, it only focuses on the economic benefits and waiting time of the vehicle and ignores the impact of fast charging station loads when charging large-scale electric vehicles. Most charging scheduling uses a fixed strategy while ignoring the influence of various factors, such as the increase in the number of electric vehicles and user habits, on electric vehicle charging scheduling for different time periods.

In this context, we propose an electric vehicle charging service scheduling method based on reinforcement learning to meet the needs of electric vehicle owners. The structure of the paper is as follows. In Section 2, we propose a fast charging station and electric vehicle system framework and use this framework to study electric vehicle charging navigation. In Section 3, we establish a dynamic travel time model for traffic sections and propose an electric vehicle charging navigation model. In Section 4, incorporating reinforcement learning, we further propose an electric vehicle charging navigation scheduling method to rationally optimize the service resources of each fast charging station in the area. In Section 5, we use a certain city as a model and compare the simulation results of the proposed method with those of the traditional electric vehicle charging navigation method to demonstrate the superiority of this method. Conclusions and further research directions are outlined in Section 6.

2. Fast Charging Station and Electric Vehicle System Framework

With the gradual development and application of 4G and 5G communications, the applications of various technologies for navigation and vehicle-road collaboration have become increasingly mature [15, 16]. At the same time, edge computing technology also provides technical guarantees for fast response and low error rate operating environments. The computational burden of the central scheduling node is transferred to the user edge side, which greatly increases the processing efficiency and enables electric vehicles and fast charging stations to share information and synchronize processing [17].

Currently, electric vehicles can share information with fast charging stations and other systems through the Internet, upload the status and location of electric vehicles in real time, and navigate in real time based on the location of electric vehicles [18, 19]. Moreover, a variety of optimal dynamic charging methods for electric fleets based on adaptive learning have been proposed, and the results show that this method can basically achieve the optimal solution. On this basis, the optimal route charging schedule can be effectively carried out for the electric fleet of efficient and dynamic transportation systems. Inspired by the above research, this paper proposes a guidance system structure for electric vehicles and fast charging stations. The structure of the guidance system for electric vehicles and fast charging stations in this article is shown in Figure 1. With the Internet platform as the center, the system dynamically updates intersection information and provides dynamic charging and navigation strategies for electric vehicles by referring to road condition information and fast charging station information. Navigation combines the road condition information and the waiting time of each fast charging station and chooses the fast charging station with the highest overall efficiency for itself to charge. The fast charging station itself further charges the electric vehicle according to various factors, such as weather, energy supply and demand, and user habits. At intervals, the traffic information and fast charging station information are refreshed according to the above selection, and the charging navigation strategy is provided again.

3. Preliminary Model Establishment

This section first proposes a dynamic travel time model for traffic sections and, on this basis, establishes a charging navigation model that considers distance, time, and economic benefits for a single electric vehicle.

3.1. Dynamic Travel Time Model of Traffic Section. The dynamic path selection model for electric vehicles in this paper is based on the dynamic travel time model of the road segment. First, the movement of the vehicle in the road segment is described by the cumulative number of vehicles $M(a,t)$, which represents the sum of the number of vehicles passing observation point a before time t. According to the definition of flow and density, the traffic flow $\sigma(a,t)$ and traffic density $\rho(a,t)$ are as follows:

$$\begin{aligned}\sigma(a,t) &= \lim_{t\longrightarrow t_0}\left(\frac{M(a,t)-M(a,t_0)}{t-t_0}\right)=\frac{\partial M(a,t)}{\partial t},\\ \rho(a,t) &= \lim_{a\longrightarrow a_0}\left(\frac{M(a_0,t)-M(a,t)}{a-a_0}\right)=\frac{\partial M(a,t)}{\partial a},\end{aligned} \tag{1}$$

where $M(a,t_0)$ and $M(a_0,t)$ are the number of vehicles at position a at time t_0 and the number of vehicles at position a_0 at time t, respectively.

According to the traffic volume and traffic density, the traffic velocity $v(a,t)$ can be obtained as follows:

$$v(a,t)=\frac{\sigma(a,t)}{\rho(a,t)}=\frac{\partial M(a,t)}{\partial t}\cdot\frac{\partial a}{\partial M(a,t)}. \tag{2}$$

Assuming that the vehicles on the road section are evenly distributed in the road section, the traffic density $\rho_i(t)$ of road section i is as follows:

$$\rho_i(t)=\frac{M\left(a_i^0,t\right)-M\left(a_i^L,t\right)}{L_i\cdot n_i}, \tag{3}$$

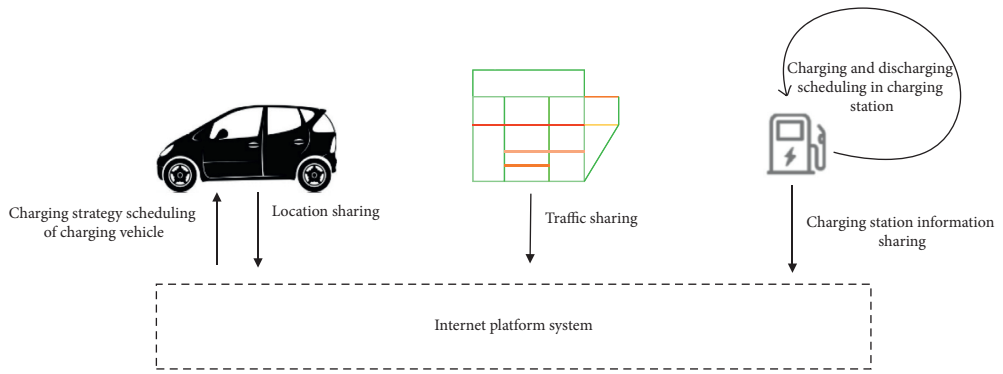


Figure 1: Electric vehicle and charging station guidance system structure.

where a_i^0 and a_i^L are the entrance and exit positions of road section i, respectively; n_i is the number of vehicles that can be accommodated per unit length in road section i; and L_i is the length of road section i.

According to the above formula, the vehicle speed $v_i(t)$ on road section i can be expressed as follows [20]:

$$v_i(t) = \begin{cases} v_{i,\text{free}}, & \rho_i(t) < \rho_{i,\min}, \\ v_{i,\min} + \left(v_{i,\text{free}} - v_{i,\min}\right)\left[1 - \left(\dfrac{\rho_i(t) - \rho_{i,\min}}{\rho_{i,\max} - \rho_{i,\min}}\right)^{\alpha}\right]^{\beta} & \rho_{i,\min} \le \rho_i(t) \le \rho_{i,\max}, \\ v_{i,\min}, & \rho_i(t) > \rho_{i,\max}, \end{cases} \tag{4}$$

where $v_{i,\text{free}}$ is the free flow velocity of section i; $\rho_{i,\max}$ and $\rho_{i,\min}$ are the maximum density and minimum density on section i, respectively; $v_{i,\min}$ is the minimum vehicle speed; and α and β are system model parameters.

It can be concluded that the passing time T_i of road section i is expressed as follows:

$$T_i = \frac{L_i}{v_i(t)} = \frac{L_i}{(\partial M(a,t)/\partial t) \cdot (\partial a/\partial M(a,t))}. \tag{5}$$

If the road congestion signal is received halfway, the system changes the route to reduce the delay time. The subjective probability of the owner changing road section i to road section i' is $P_{i \longrightarrow i'}$:

$$P_{i \longrightarrow i'} = \frac{\int_0^{T_i} e^{-\left(\left(T_{Max} - T_{i'}\right)^2 / 2(\eta)^2\right)} \mathrm{d}(T_{\text{Max}})}{\sqrt{2\pi}\eta}, \tag{6}$$

where T_i is the travel time of section i in the route; $T_{i'}$ is the travel time of section i' in the route; T_{Max} is the maximum travel time; and η is a subjective coefficient.

Therefore, the length of the driving section can be approximated by subjective probability as d_i:

$$\begin{aligned} d_i &= \left(1 - P_{i \longrightarrow i'}\right) L_i + P_{i \longrightarrow i'} L_{i'} \\ &= \left[1 - \frac{\int_0^{T_i} e^{-\left(\left(T_{Max} - T_{i'}\right)^2 / 2(\eta)^2\right)} \mathrm{d}(T_{\text{Max}})}{\sqrt{2\pi}\,\eta}\right] \cdot L_i + \frac{\int_0^{T_i} e^{-\left(\left(T_{Max} - T_{i'}\right)^2 / 2(\eta)^2\right)} \mathrm{d}(T_{\text{Max}})}{\sqrt{2\pi}\eta} \cdot L_{i'}, \end{aligned} \tag{7}$$

where $L_{i'}$ is the length of road section i'.

3.2. Electric Vehicle Charging Navigation Model. Electric vehicles need to be charged frequently during use, so there will be demand for fast charging. According to the charging needs of different vehicles, implementing different navigation schemes can effectively improve the response speed of the vehicle. This section comprehensively considers the driving distance required to reach a fast charging station, the total time of driving and charging, and the charging economy to establish a charging navigation model.

For electric vehicle owners with high total driving distance requirements, this article considers the principle that the direction of the fast charging station is the same as the destination direction when all vehicles are connected to the Internet. It is proposed that the sum of the shortest distance min D from the starting point O of the vehicle to the fast charging station S and from the fast charging station S to the destination D is expressed as follows:

$$\min D = \sum_{a=1}^{m} \sum_{b=1,b\neq a}^{m} d_{ab,OS}\alpha_{ab} + \sum_{a=1}^{m} \sum_{b=1,b\neq a}^{m} d_{ab,S\ D}\alpha_{ab}, \quad (8)$$

where a and b are path nodes; m is the total number of path nodes; $d_{ab,OS}$ and $d_{ab,S\ D}$ are the length of the road section from the starting point O to the fast charging station S and from the fast charging station S to the destination D with a and b as the end nodes; and α_{ab} is a variable that equals 1 for the road section with a and b as the end nodes and equals 0 otherwise.

For electric vehicle owners with high total time requirements, this article proposes the shortest total charging time as the goal to optimize the charging path:

$$\min T = T_D + T_C + T_Q. \quad (9)$$

The specific solutions of T_D and T_C are as follows:

$$\begin{aligned} T_D &= \sum_{i=1}^{i=m} \frac{d_i}{v_i(t)} \\ &= \sum_{i=1}^{i=m} \frac{(1-P_{i\longrightarrow i'})L_i + P_{i\longrightarrow i'}L_{i'}}{v_i(t)} \\ &= \sum_{i=1}^{i=m} \frac{(1-P_{i\longrightarrow i'})L_i + P_{i\longrightarrow i'}L_{i'}}{(\partial M(a,t)/\partial t)\cdot(\partial a/\partial M(a,t))}, \end{aligned} \quad (10)$$

$$\begin{aligned} T_C &= \frac{Q_{\mathrm{Ex}} - Q_{\mathrm{Re}}}{P\theta} \\ &= \frac{Q_{\mathrm{Ex}} - \left(C_{\mathrm{car}}\cdot C_{\mathrm{carINI}} - Q\int_0^{T_D} v_i(t)\mathrm{d}t\right)}{P\cdot\theta}, \end{aligned}$$

where T_D is the travel time to the fast charging station; T_Q is the waiting time in the fast charging station, which is determined by the number of vehicles; T_C is the charging time; Q_{Ex} is the expected voltage at the end of charging, which is set to 95% of the full charge; Q_{Re} is the remaining power to the fast charging station; P is the charger power; θ is the charging efficiency; C_{car} is the electric vehicle battery capacity; C_{carINI} is the initial state of charge of the electric vehicle; and Q is the electric energy consumed by the electric vehicle per kilometer.

For electric vehicle owners with high cost requirements, this article proposes the minimum cost as the goal to optimize the charging path:

$$\min M = M_D + M_S, \quad (11)$$

where M_D is the electricity cost consumed on the charging path and M_s is the cost consumed by the fast charging station.

4. Electric Vehicle Charging Navigation Scheduling Strategy Based on Reinforcement Learning

The goal of the reinforcement learning algorithm is to find an optimal strategy based on the Markov decision process to maximize the expected cumulative return. In this section, the driving distance of the electric vehicle, the total driving and charging time, and the charging economy are optimized in parallel to provide the electric vehicle owner with the best electric vehicle charging navigation scheduling strategy [21, 22].

4.1. Strategy Gradient Algorithm. The basic principle of reinforcement learning is to learn from exploratory experiments and obtain action strategies to achieve established goals. The learning subject is the agent; the object interacting with the agent is the environment. Reinforcement learning is an abstraction of goal-oriented interactive learning problems. In a certain environment state, the agent takes action, and the environment responds to the agent's actions, presents the new environment state to the agent, and feeds a certain reward back to the agent. The agent and the environment continue to interact to achieve the ultimate goal of maximizing returns.

The interaction process between the agent and the environment can be described by a time series: in a certain period t, the agent takes a certain action a according to the current environment state s_t^n; in the next period $t+1$, due to the agent's action a_t^n, the environment state changes from s_t^n to s_{t+1}^n, and the agent is rewarded with $r(t)^n$. In each time period, the probability distribution of all actions that the agent can take in the current environment state is called the agent's strategy π. The agent continuously changes its strategy through interaction and finally achieves the goal of maximizing rewards.

The reinforcement learning problem satisfies the Markov characteristic; that is, the state of the next period is only related to the state s_t^n of the current period and has nothing to do with the state s_{t+1}^n of the previous period. The policy-based method is used to express a policy. Assuming that the strategy of electric vehicle charging and navigation control consists of a t-step decision, the agent obtains n corresponding training trajectories τ_n by interacting with the environment as follows:

$$\tau_n = [s_1^n, r(1)^n, a_1^n, s_2^n, r(2)^n, a_2^n, \ldots, s_t^n, r(t)^n, a_t^n], \quad (12)$$

where a_t^n represents the action determined at time t during the n training, s_t^n represents the state after action a during the n training, and $r(t)^n$ represents the reward obtained after action a during the n training. The expected return reward R_θ for all stored trajectories is as follows:

$$R_\theta = \sum_{n=1} R(\tau_n) \bullet p_\theta(\tau_n) = \sum_{n=1} R(\tau_n) \bullet \prod_{t=1}^{T} [p(a_t^n | s_t^n, \theta) \cdot p(r(t+1)^n, s_{t+1}^n | s_t^n, a_t^n)], \tag{13}$$

where $R(\tau_n) = \sum_{i=0}^{t} r(t)^n$ is the reward value of trajectory τ_n, $p_\theta(\tau_n)$ is the probability of trajectory τ_n, $p(r(t+1)^n, U_{t+1}^n, f_{t+1}^n | U_t^n, f_t^n, a_t^n)$ is the probability of $[r(t+1)^n, s_{t+1}^n]$ in state $[s_t^n, a_t^n]$, and $p(a_t^n | s_t^n, \theta)$ is the probability of selecting actions a_t^n according to input and output strategy $\pi_n(\theta)$ in state s_t^n.

Therefore, reinforcement learning can be expressed as solving the maximum expected return reward R_θ. To realize the strategy, the partial derivative of the parameter set θ is obtained to obtain the optimized strategy function ∇R_θ as follows:

$$\begin{aligned}
\nabla R_\theta &= \nabla \sum_{n=1} R(\tau_n) \bullet p_\theta(\tau_n) \\
&= \sum_{n=1} R(\tau_n) \bullet \nabla p_\theta(\tau_n) \\
&= \sum_{n=1} R(\tau_n) \bullet p_\theta(\tau_n) \frac{\nabla p_\theta(\tau_n)}{p_\theta(\tau_n)} \\
&= \sum_{n=1} R(\tau_n) \bullet p_\theta(\tau_n) \bullet \nabla \log p_\theta(\tau_n) \\
&= \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) \bullet \nabla \log p_\theta(\tau_n) \\
&= \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) \bullet \nabla \log \left\{ \prod_{t=1}^{T} [p(a_t^n | s_t^n, \theta) \bullet p(r(t)^n, s_{t+1}^n | s_t^n, a_t^n)] \right\} \\
&= \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) \bullet \nabla \left\{ \sum_{t=1}^{T} [\log p(a_t^n | s_t^n, \theta) + p(r(t)^n, s_{t+1}^n | s_t^n, a_t^n)] \right\} \\
&= \frac{1}{N} \sum_{n=1}^{N} R(\tau_n) \bullet \nabla \sum_{t=1}^{T} \log p(a_t^n | s_t^n, \theta) \\
&= \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T} R(\tau_n) \bullet \nabla \log p(a_t^n | s_t^n, \theta).
\end{aligned} \tag{14}$$

The reinforcement learning policy gradient algorithm is equivalent to solving a partial derivative problem. If the parameter set θ is updated in the positive direction, that is, the reward increases, the probability of trajectory τ_n will increase, and vice versa. The pseudocode of the policy gradient Algorithm 1 is given below.

4.2. Action Selection. Taking the vehicle travel path as an example, the control parameter is a_t^n, and the vehicle has 3 possible actions at each intersection. The value range is $a_t^n = [0, 2]$. 0 means going forward, 1 means turning left, and 2 means turning right.

4.3. Environmental Status. When an electric vehicle performs an action at an intersection and acts on the environment, the state value corresponding to the environmental feedback is s_t^n, that is, $s_t^n = [D_t^n, T_t^n, M_t^n]$, and the dimension is 3. Among them, D_t^n, T_t^n, and M_t^n correspond to the distance, time, and cost, respectively, after the current action is executed. After obtaining the environmental state value, the corresponding reward value is calculated, and at the same time, the environment will move to the next state.

4.4. Reward Function Design. The reward function is designed as follows:

$$\text{reward} = -a \cdot (D_t^n - \min D)^2 - b \cdot (T_t^n - \min T)^2 - c \cdot (M_t^n - \min M)^2, \tag{15}$$

where reward represents the reward value obtained by the action performed by the electric vehicle at each time node, that is, the quality of the current trajectory τ_n action.

(1) In the neural network, initialize the parameter set θ randomly and initialize $n = 1$.
(2) Initialize $t = 0$, randomly initialize action a_t^n and output state s_{t+1}^n, calculate local reward $r(t+1)^n$, and then add the trajectory generated by the $t + 1$ action to the stored trajectory τ_n of the n training.
(3) Input state s_{t+1}^n to the neural network and select a random action a_{t+1}^n.
(4) After the simulation environment executes action a_{t+1}^n, obtains the output state s_{t+2}^n, and calculates the local reward $r(t+2)^n$, the trajectory generated by the $t + 2$ action is added to the stored trajectory τ_n of the n training.
(5) Judge whether $R(\tau_n) = \sum_{i=1}^{i=t} r(t)^n > R$ is true; if it is true, go to step 6; otherwise, assign $t + 1$ to t and go to step 3, where i is the variable to be accumulated and R is the expected value of the total reward for a single trajectory.
(6) Calculate the strategy optimization strategy function ∇R_θ.
(7) Assign $n + 1$ to n, update the parameter set θ in strategy $\pi_n(\theta)$ to $\theta + \gamma * \nabla R_\theta$, and judge whether $n \leq N$ is true; if so, go to step 2; otherwise, the reinforcement learning training process is over; save the updated parameter set as the most optimal parameter set θ^* and the optimal strategy π; N is the maximum number of trajectories

Algorithm 1: Policy gradient algorithm

Among them, a, b, and c are the weighting coefficients: when the owner only cares about the distance, a equals 1, and the rest equal 0; when the owner only cares about the total time, b equals 1, and the rest equal 0; when the owner only cares about the cost, c equals 1, and the rest equal 0; and if the owner chooses to focus on all three variables, set $a + b + c = 1$, and assign values according to the proportion.

4.5. Controller. For electric vehicle charging navigation, a scheduling algorithm based on the policy gradient algorithm is proposed according to the personal habits of different electric vehicle owners. By observing the information to select a behavior directly for back propagation and using rewards to directly enhance and weaken the possibility of selection behavior, the probability of selecting good behavior will increase next time, and bad behavior will be weakened next time.

A three-layer wavelet neural network is used. The wavelet neural network is a multilayer feedforward neural network trained according to error back propagation [23]. This article uses a three-layer neural network, that is, one output layer, one input layer, and one hidden layer, as shown in Figure 2. The state is set as the input layer of the neural network. Its dimension is 3; the hidden layer of the neural network has 20 neurons; and the output layer contains 3 neurons, corresponding to 3 output actions.

The connection weights and bias terms between the input layer and the hidden layer and between the hidden layer and the output layer are represented by a parameter set of θ. The input and output strategies of the n training wavelet neural network of the strategy body are defined as $\pi_n(\theta)$.

The activation function of the connection between the input layer and the hidden layer is $\mathrm{Tan}\,h$, and its function formula is as follows:

$$f(z) = \mathrm{Tan}\,h(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}. \tag{16}$$

The activation function connecting the hidden layer and the output layer is a wavelet basis function, and its function formula is as follows:

$$\varsigma(z) = e^{-\left(z^2/z\right)} \cos(1.75z). \tag{17}$$

According to the pseudocode of the algorithm, the specific training process can be obtained as shown in Figure 3.

5. Simulation Results and Discussion

Taking the city in Figure 4 as a model, the city includes 21 nodes, 32 road sections, and 4 fast charging stations. The number marked on the road section represents the length of the road section in km. Fast charging stations are located at nodes 9, 12, 14, and 19. For electric vehicles, the battery capacity is 90 kW·h, the cruising range is 400 km, and the fast charging station power is 350 kW. When the electric vehicle leaves the fast charging station, C_{carINI} is 90%; the training parameters are as follows: the number of training rounds is 1900, and the learning coefficient is 0.95. The discount rate is 0.95.

The vehicle randomly sets the initial position and target position (on 21 nodes) and randomly sets the remaining power (not higher than 30%). According to the distance selected by the user, the total time consumed, and the cost as the reward value, the vehicle is trained from the initial position to the fast charging station to charge and from the fast charging station to the target location. After the training is completed, the final reward changes are shown in Figure 5.

Figure 5 shows that as the number of training sessions increases, the training reward gradually increases. After 600 training sessions, the curve shows an oscillating trend, and the reward oscillates around 190. In the subsequent training, the reward is basically stable. Save the neural network model obtained from the last training parameter.

The 08:00 traffic flow distribution obtained through urban traffic simulation is shown in Figure 6. The green line represents smooth traffic, orange represents traffic congestion, and red represents heavy traffic congestion. For the traffic flow shown in Figure 6, the saved reinforcement learning model is used to obtain the station selection probability of the electric vehicle when each network node starts, as shown in Figure 7. It can be concluded that under the premise of considering congestion, the trained reinforcement learning model can effectively select fast charging

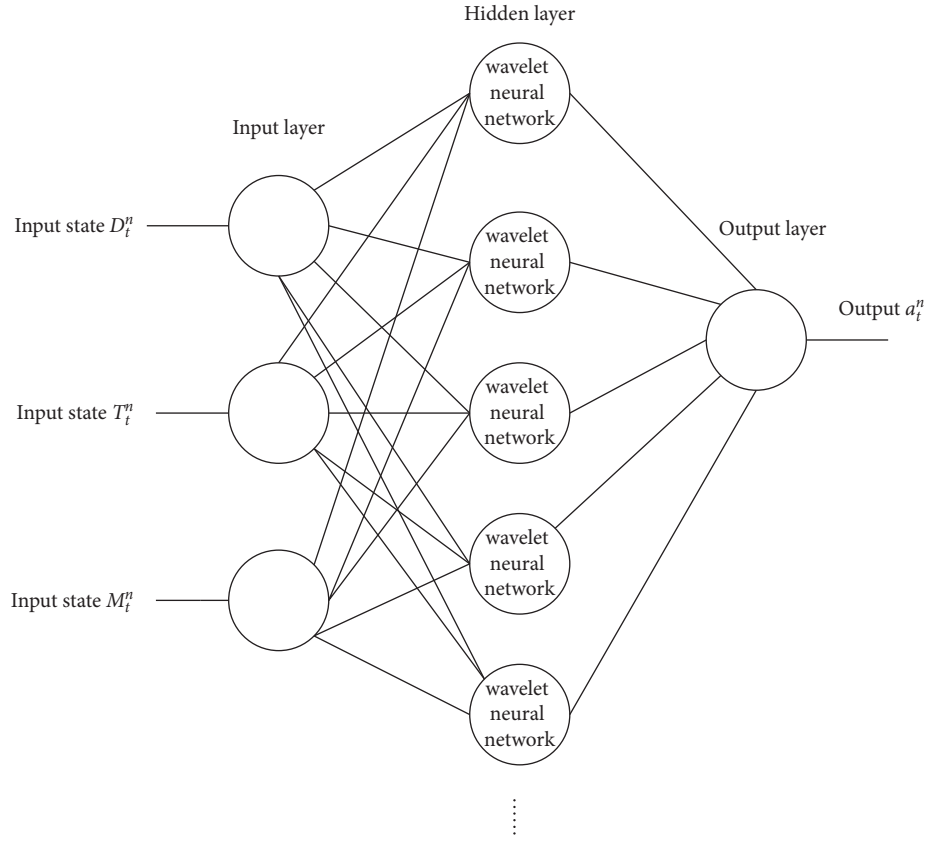


Figure 2: Structure diagram of wavelet neural network.

stations corresponding to shorter distances according to the target node.

Now, take an electric vehicle starting at node 13 and ending at node 2 as an example to analyze its dynamic station selection strategy. Consider the distance, total time, and cost required for the owner to obtain charging navigation during driving, as shown in Table 1.

Plan 1 takes the minimum distance as the goal and chooses fast charging station No. 9, and the travel route is shown as the solid line in Figure 8. Plan 2 takes the minimum time as the goal and chooses fast charging station No. 14, and the travel route is shown by the dashed line in Figure 8. Plan 3 takes the minimum cost as the goal and chooses No. 12 fast charging station, and the travel route is shown as the crossed line in Figure 8.

Multiple routes were selected for testing, and methods from [10, 13] and the charging navigation method proposed in this paper were compared. The performance comparisons under the comprehensive requirements of the research vehicle owners are shown in Figure 9.

The first graph in Figure 9 shows the change trend of the average distance with the increase in the number of test routes under the premise of considering the comprehensive performance required by the user. In this graph, the comparison between the method in this paper and the methods in the other two references is shown. With the increase in the number of routes, the average travel distance of the three methods fluctuated and finally stabilized in the vicinity of 17 km. In this process, the total distance predicted by the three methods is basically the same. The second graph in Figure 9 shows the trend of the total time as the number of test routes increases. As the number of routes increases, the total time of the method in this paper steadily decreases, and finally the time is reduced to 0.7 h, while for the other two methods, the total time consumed curve presents an oscillating situation, and the time consumed is unstable and greater than that for the method in this paper. It can be concluded from the curve that the method in this paper has the least total time consumption. The third graph in Figure 9 shows the trend of the total cost as the number of test routes increases. With the increase in the number of routes, the total cost of the method in this paper first increases, then gradually decreases, and finally stabilizes at approximately 30 yuan. For the method from [10], the total cost of the method was initially lower than that of the method in this paper. With the increase in the number of test routes, the cost began to increase and eventually was significantly higher than that of the method in this paper. The cost for the method in [13] remained higher than the cost for the method in this paper after initially oscillating lower. It can be concluded that, under the comprehensive performance requirements, the total distances of the three methods are basically the same. On this basis, with the increase in the number of route tests, the method in this paper has the least total time and cost, which indicates the superiority of the method in this paper.

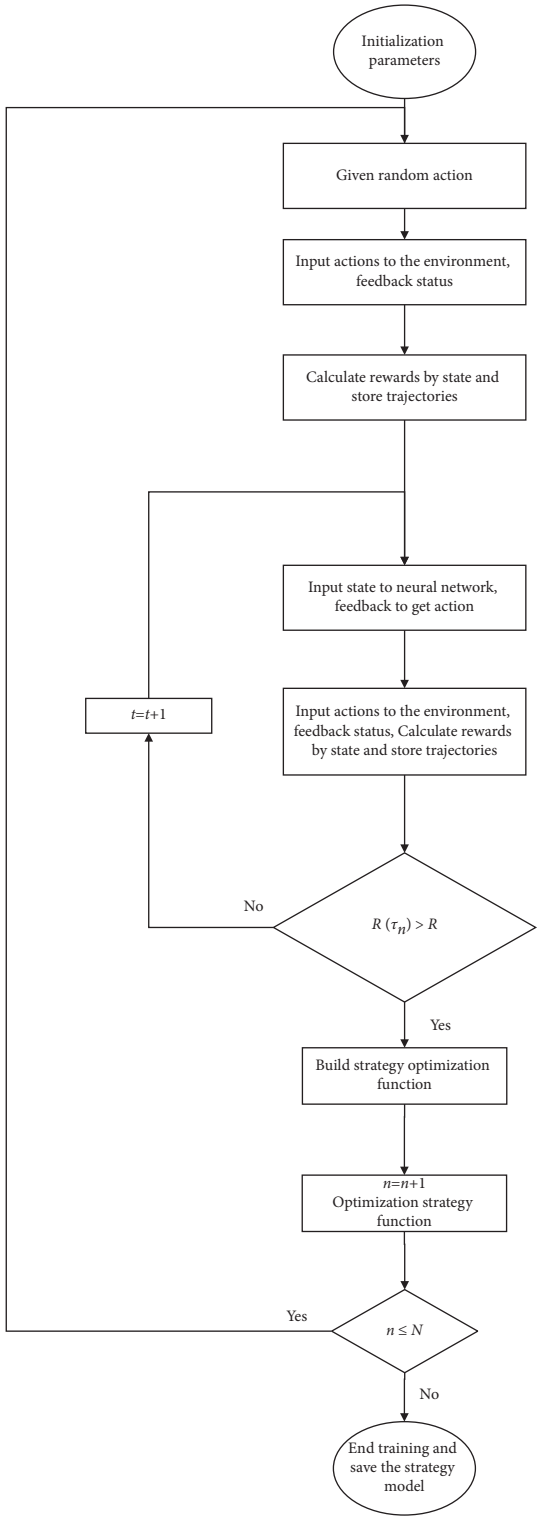


Figure 3: Flowchart of training steps.

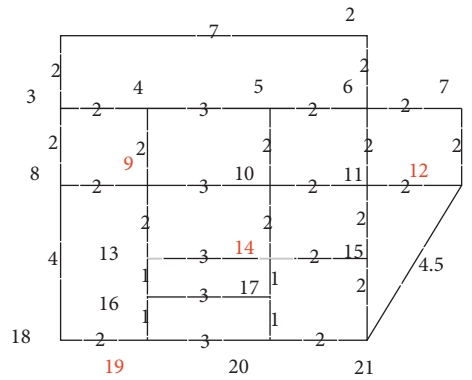


Figure 4: City model chart.

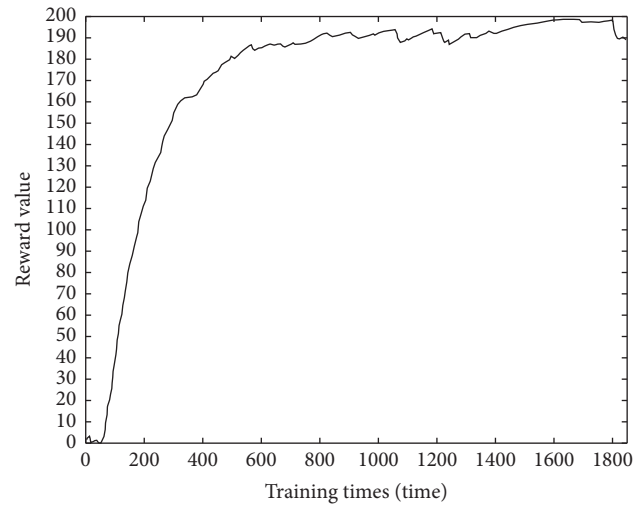


Figure 5: Training result chart.

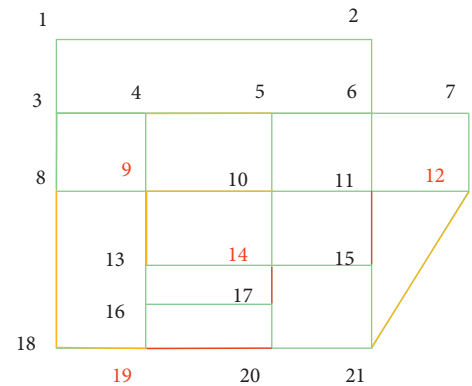


Figure 6: Traffic flowchart.

In the case of the same time, initial point, and destination, we compare user satisfaction under the electric vehicle charging navigation strategy in [10, 13, 24–26]. The user satisfaction from testing electric vehicles using these methods is shown in Table 2. It can be seen in the table that, with the increase of test lines, user satisfaction under this

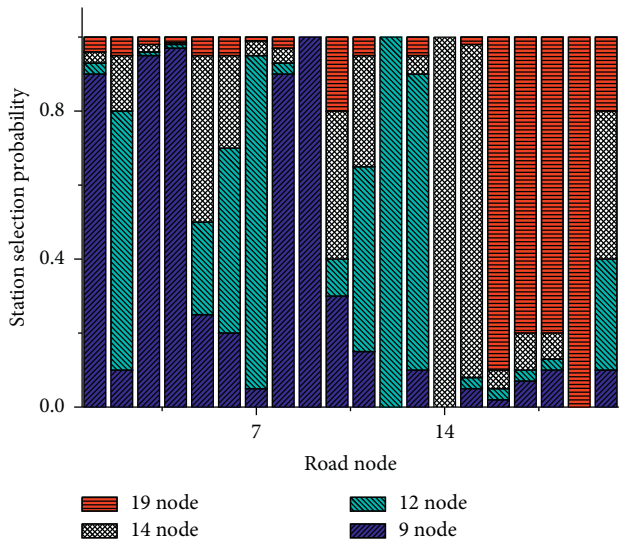


Figure 7: Probability of selecting a fast charging station node.

Table 1: Scheme comparison.

Plan	Distance (km)	Total time (h)	Expenses (yuan)	Station selection
1	11	1.5	76	9
2	11	1	84	14
3	15	1.8	60	12

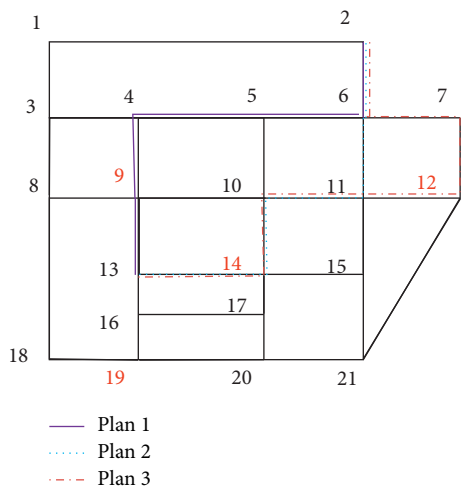


Figure 8: Route selection chart.

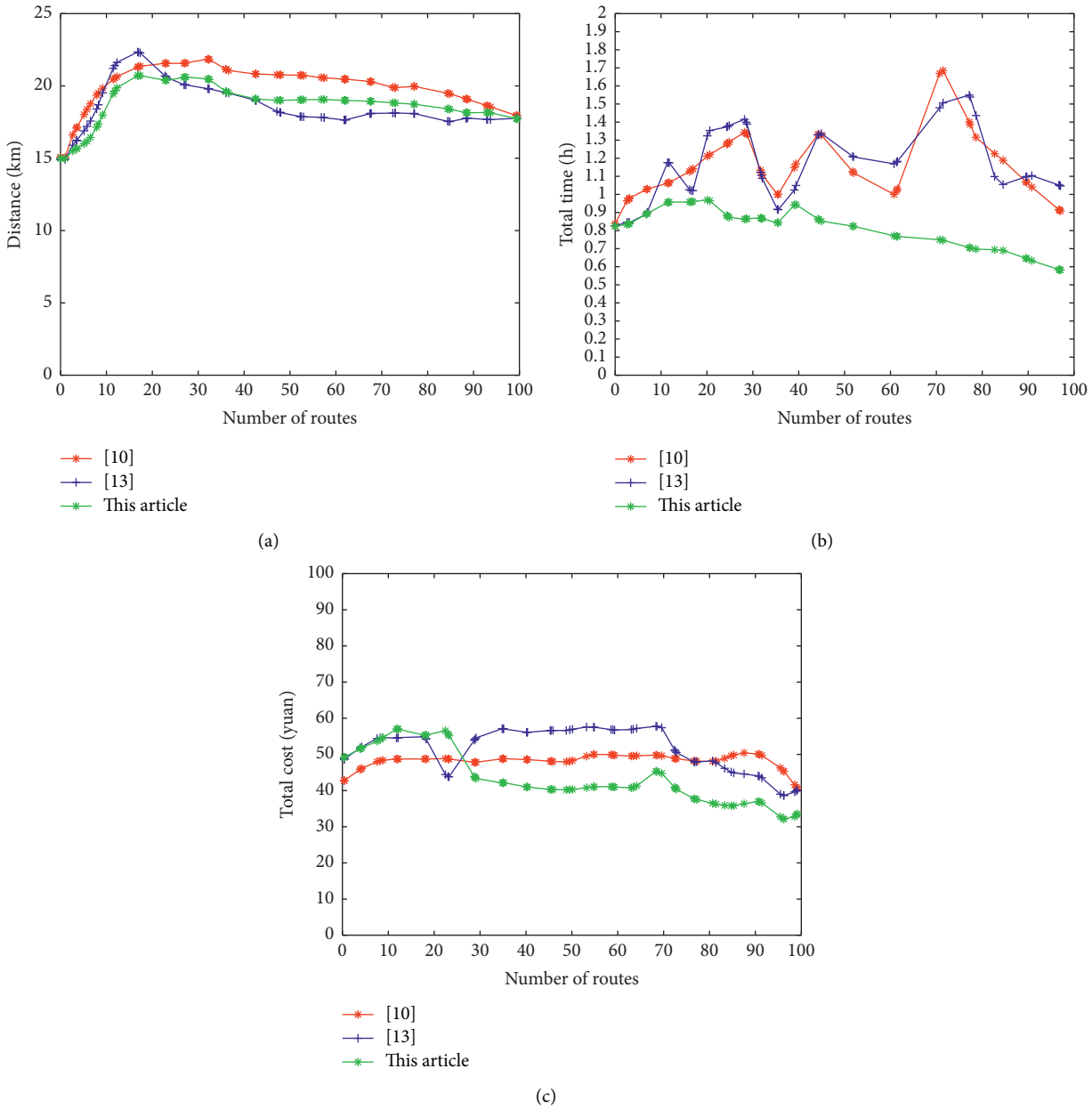


Figure 9: Distance, time, and cost comparison chart.

Table 2: User satisfaction for electric vehicle charging navigation strategies.

Number of routes	This article (%)	[10] (%)	[13] (%)	[24] (%)	[25] (%)	[26] (%)
1	100	100	100	100	100	100
10	100	95	80	90	90	70
100	95	80	70	90	80	75
1000	90	75	75	85	75	55

method far exceeds other methods. It has been demonstrated that the method in the article can effectively meet the charging and navigation needs of users.

6. Conclusions

We propose an electric vehicle charging service scheduling method based on reinforcement learning to meet the needs of electric vehicle owners. First, based on an intelligent transportation system, a framework for the interaction between fast charging stations and electric vehicles is proposed. Subsequently, the dynamic travel time model of the traffic section was established, and the electric vehicle charging navigation model was proposed. Finally, combined with reinforcement learning, the electric vehicle charging navigation scheduling method is further proposed to rationally optimize the service resources of each fast charging station in the area. The results show that, compared with the existing methods, the algorithm and model proposed in this paper can effectively optimize electric vehicle charging and navigation scheduling based on the needs of the vehicle owner and can meet the various needs of the vehicle owner.

Data Availability

The MATLAB simulation data used to support the findings of this study are currently under embargo while the research findings are commercialized. Requests for data, 12 months after publication of this article, will be considered by the corresponding author.

Conflicts of Interest

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

References

[1] X. Sun, Z. Li, and X. Wang, "Technology development of electric vehicles: a review," *Energies*, vol. 90, 2020.

[2] A. V. Prakash and S. Parameswaran, "Smart parking and charging management of electric vehicles in public parking space," in *Proceedings of the 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, 2018.

[3] S. H. Cutcliffe and D. A. Kirsch, "The electric vehicle and the burden of history," *History*, vol. 6, no. 2, p. 326, 2000.

[4] K. Coninx, R. Claes, and S. Vandael, "Anticipatory coordination of electric vehicle allocation to fast charging infrastructure," in *Advances in Practical Applications of Heterogeneous Multi-Agent Systems*Springer, Berlin, Germany, 2014.

[5] P. Morrissey, P. Weldon, and M. O'Mahony, "Future standard and fast charging infrastructure planning: an analysis of electric vehicle charging behaviour," *Energy Policy*, vol. 89, pp. 257–270, 2016.

[6] M. Yang and Zhang, "The time-of-use price of electric vehicle charging and discharging based on monte carlo algorithm," in *Proceedings of the 2018 International Conference on Big Data and Artificial Intelligence (ICBDAI 2018)*, 2018.

[7] F. Liu, "Applying fuzzy logic control to analyze real-time control for charging and discharging power of electric vehicles," *Journal of Intelligent and Fuzzy Systems*, vol. 4, pp. 1–7, 2021.

[8] F. Ni, Z. J. Liu, H. Zhang, L. Shuo, and W. Feng, "Review of research on coordinated charging of electric vehicles," *Advanced Materials Research*, vol. 860-863, pp. 1164–1172, 2014.

[9] M. T. Hussain, N. B. Sulaiman, M. S. Hussain, and J. Muhammad, "Optimal management strategies to solve issues of grid having electric vehicles (EV): a review," *The Journal of Energy Storage*, vol. 33, no. 3, Article ID 102114, 2020.

[10] S. Pourazarm, C. G. Cassandras, and A. Malikopoulos, "Optimal routing of electric vehicles in networks with charging nodes: a dynamic programming approach," in *Proceedings of the 2014 IEEE International Electric Vehicle Conference (IEVC)*, December 2019.

[11] T. Wang, C. Cassandras, and S. Pourazarm, "Energy-aware vehicle routing in networks with charging stations," *IFAC Proceedings Volumes*, vol. 473, p. 9611, 9616.

[12] S. Pourazarm and C. Cassandras, "Optimal routing of energy-aware vehicle in networks with inhomogeneous charging nodes," in *Proceedings of the 22nd IEEE Mediterranean Conference on Control and Automation*, pp. 674–679, Palermo, Italy, June 2014.

[13] M. Elgarej, M. Khalifa, and M. Youssfi, "Optimized path planning for electric vehicle routing and charging station navigation systems," *International Journal of Applied Metaheuristic Computing*, vol. 11, no. 3, pp. 58–78, 2020.

[14] S. Zhang, G. Feng, Y. Fan, S. Wan, and Y. Luo, "Large-scale electric vehicle charging path planning based on information interaction," *Qinghua Daxue Xuebao/Journal of Tsinghua University*, vol. 58, no. 3, pp. 279–285, 2018.

[15] Y. Shen, W. Fang, F. Ye, and M. Kadoch, "EV charging behavior analysis using hybrid intelligence for 5G smart grid," *Electronics*, vol. 9, no. 1, p. 80, 2020.

[16] M. Y. Yen, H. C. Chen, Y. L. Wei, and Chung, "A CMOS transmitter analog baseband for 5G mobile communication," *Electronics*, vol. 8, no. 11, p. 1319, 2019.

[17] W. Qi, B. Zla, N. Ke, C. Yifan, and W. Ming, "Dynamic resource allocation for jointing vehicle-edge deep neural network inference," *Journal of Systems Architecture*, vol. 117, Article ID 102133, 2021.

[18] T. Nishimura and H. Koizumi, "A real-time traffic flow measurement method based on inter-vehicle communication considering the volume of vehicle information," 情報処理学会研究報告高度交通システム*(ITS)*, vol. 2006, pp. 1–8, 2006.

[19] R. Sun, Y. Yang, K. W. Chiang, D. Thanh-Trung, L. Kuan-Ying, and T. Guang-Je, "Robust IMU/GPS/VO integration for vehicle navigation in GNSS degraded urban areas," *IEEE Sensors Journal*, vol. 20, no. 17, p. 1, 2020.

[20] J. Tan and L. F. Wang, "Real-time charging navigation of electric vehicles to fast charging stations: a hierarchical game approach," *IEEE Transactions on Smart Grid*, vol. 8, no. 2, pp. 846–856, 2017.

[21] G. Li, N. Cao, P. Zhu et al., "Towards smart transportation system," *Journal of Organizational and End User Computing*, vol. 33, no. 3, pp. 35–49, 2021.

[22] D. Chen, Y. Wang, and W. Gao, "Combining a gradient-based method and an evolution strategy for multi-objective reinforcement learning," *Applied Intelligence*, vol. 50, no. 10, pp. 3301–3317, 2020.

[23] M. Zaher, A. Shehab, M. Elhoseny, and F. F. Farahat, "Unsupervised model for detecting plagiarism in internet-based

handwritten Arabic documents," *Journal of Organizational and End User Computing*, vol. 32, no. 2, pp. 42–66, 2020.

[24] C. Liu, W. Jing, and C. Long, "Joint charging and routing optimization for electric vehicle navigation systems," *IFAC Proceedings Volumes*, vol. 47, no. 99, pp. 2106–2111, 2014.

[25] G. Li, Q. Sun, L. Boukhatem, J. Wu, and J. Yang, "Intelligent vehicle-to-vehicle charging navigation for mobile electric vehicles via VANET-based communication," *IEEE Access*, vol. 7, pp. 170888–170906, 2019.

[26] X. Li, Y. Xiang, L. Lyu et al., "Price incentive-based charging navigation strategy for electric vehicles," *IEEE Transactions on Industry Applications*, vol. 56, no. 5, pp. 5762–5774, 2020.

Hindawi
Mathematical Problems in Engineering
Volume 2021, Article ID 6253424, 9 pages
https://doi.org/10.1155/2021/6253424

Research Article

A Class of Inexact Secant Algorithms with Line Search Filter Method for Nonlinear Programming

Zhujun Wang[1] **and Li Cai**[2]

[1]*School of Computational Science and Electronics, Hunan Institute of Engineering, Xiangtan 411105, China*
[2]*Shanghai Research Institute of Microwave Equipment, Shanghai 200063, China*

Correspondence should be addressed to Zhujun Wang; 70127@hnie.edu.cn

Received 12 September 2021; Accepted 9 October 2021; Published 10 November 2021

Academic Editor: Guoqiang Wang

We propose a class of inexact secant methods in association with the line search filter technique for solving nonlinear equality constrained optimization. Compared with other filter methods that combine the line search method applied in most large-scale optimization problems, the inexact line search filter algorithm is more flexible and realizable. In this paper, we focus on the analysis of the local superlinear convergence rate of the algorithms, while their global convergence properties can be obtained by making an analogy with our previous work. These methods have been implemented in a Matlab code, and detailed numerical results indicate that the proposed algorithms are efficient for 43 problems from the CUTEr test set.

1. Introduction

This paper considers the following nonlinear equality constrained optimization problem:

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} f(x), \\ &\text{s.t.} c(x) = 0, \end{aligned} \tag{1}$$

where $f\colon \mathbb{R}^n \longrightarrow \mathbb{R}$ and $c\colon \mathbb{R}^n \longrightarrow \mathbb{R}^m$ are twice continuously differentiable.

After more than 20 years of development, filter algorithm has become an important method to solve constrained optimization problems and has been successfully applied to various fields of optimization (see [1–5]). In [6, 7], a line search filter Newton method is presented for (1). Local and global convergence properties of this method are analyzed. In particular, motivated by a multiobjective restructuring of constrained optimization problems, Eichfelder et al. [3] proposed a new filter approach, which utilizes the restructuring to identify the trade-off between constraint satisfaction and objective value. It is foreseeable that the filter method will find application in more diverse fields.

It is well known that the secant methods (two-step algorithms) are one of the most successful methods for solving problem (1) and have a main advantage which rests in the use of an orthogonal projection operator which is continuous. Fontecilla [8] showed that the six secant algorithms generate a sequence converging 2-step superlinearly under some suitable conditions.

When solving large-scale nonlinear programming problems or system of nonlinear equations, it would take too much time to get an exact solution in each iteration. An inexact Newton method was proposed in [9] to solve large-scale system of nonlinear equations. Subsequently, an inexact SQP algorithm was presented in [2] to solve large-scale equality constrained optimization problems. Under suitable

assumptions, the global convergence of inexact SQP steps was proved. These research results show that the use of inexact algorithm can save a lot of computing time.

Inspired by these ideas above, we propose a class of inexact secant methods in association with the line search filter technique, which has both global convergence and q-superlinear local convergence rate. These methods are globalized by line search and filter methods. In this work, we use Lagrangian function $\mathscr{L}$ as the objective function and $\|c(x)\|^2$ as the constraint violation. The filter method can suffer from the Maratos effect. The reason that Maratos effect appears is that the merit function used to judge whether the iteration point is good or bad is nonsmooth. If the merit function is smooth, Maratos effect can be avoided. Hence, we do not need to compute the second-order correction step which was used to overcome the Maratos effect.

The structure of this article is arranged as follows. Section 2 develops the algorithms. In Section 3, we mainly analyze the local superlinear convergence rate. The results of numerical experiences with the three algorithms are discussed in Section 4. Final conclusions are provided in Section 5.

2. Algorithms

In this paper, we denote the Euclidean vector or matrix norm by $\|\cdot\|$, $A(x) := \nabla c(x)$, and $g(x) := \nabla(x)$. The inexact secant algorithms are given by the following iterative scheme:

$$\lambda_{k+1} = U(x_k, \lambda_k, W_k), \tag{2a}$$

$$W_k \widehat{u}_k = -\nabla_x \mathscr{L}(x_k, \lambda_{k+1}) + r_k, \tag{2b}$$

$$u_k = P_k \widehat{u}_k, \tag{2c}$$

$$v_k = -A_k^{\dagger} c_k, \tag{2d}$$

$$d_k = u_k + v_k, \tag{2e}$$

$$y_k = \nabla_x \mathscr{L}(x_k + u_k, \lambda_{k+1}) - \nabla_x \mathscr{L}(x_k, \lambda_{k+1}), \tag{2f}$$

$$W_{k+1} = B(u_k, y_k, W_k), \tag{2g}$$

$$x_{k+1} = x_k + d_k, \tag{2h}$$

where $\|r_k\| \le \eta_k \|c_k\|$ with $\eta_k \in [0, t]$ and $t \in (0, 1)$, $\mathscr{L}(x, \lambda) = f(x) + \lambda^T c(x)$ for $\lambda \in \mathbb{R}^m$, and B is the BFGS or the DFP secant update formula generating matrices W_k approximating the Hessian of the Lagrangian at each x_k. There are many different multiplier updates; however, λ_{k+1} in (2a) is chosen from the most commonly used multiplier updates: projection update, null-space update, and Newton update. $P(x)$ in (2c) can be either the orthogonal projection or the oblique projection depending on the choice of the multiplier update λ_{k+1}. $A_k^{\dagger}$ in (2d) is the pseudo-inverse of A_k^T. With different choices for λ_{k+1} in (2a), $P(x)$ in (2c), and $A_k^{\dagger}$ in (2d), we get the following three algorithms:

(i) ALG 1: $\lambda_{k+1} = -(A_k^T W_k^{-1} A_k)^{-1} A_k^T W_k^{-1} g_k$ (null-space update)
$P(x) = I$ (I is the identity matrix)
$A_k^{\dagger} = W_k^{-1} A_k (A_k^T W_k^{-1} A_k)^{-1}$

(ii) ALG 2: $\lambda_{k+1} = -(A_k^T A_k)^{-1} A_k^T g_k$ (projection update)
$P(x) = I - W^{-1} A(x) [A(x)^T W^{-1} A(x)]^{-1} A(x)^T$ (oblique projection)
$A_k^{\dagger} = A_k (A_k^T A_k)^{-1}$

(iii) ALG 3: $\lambda_{k+1} = (A_k^T W_k^{-1} A_k)^{-1} (c_k - A_k^T W_k^{-1} g_k)$ (Newton update)
$P(x) = I - A(x)[A(x)^T A(x)]^{-1} A(x)^T$ (orthogonal projection)
$A_k^{\dagger} = W_k^{-1} A_k (A_k^T W_k^{-1} A_k)^{-1}$

To get the next iterate, the line search filter method described in Section 4.1 of the literature [6] is used to determine a step size $\alpha_k \in (0, 1]$. The Lagrangian function $\mathscr{L}(x, \lambda)$ measures optimality and the constraint violation $h(x)$ measures feasibility, where $h(x) = 1/2\|c(x)\|^2$. The class of complete algorithms can be stated as follows.

3. Convergence Analysis

The global convergence analysis of the algorithms in Section 3 of [10] still holds. The only point that needs special attention is the proof of Lemma 3.2 in [10]. This lemma proves that the search direction is a sufficient descent direction for $\mathscr{L}$at the points that are sufficiently close to feasible and nonoptimal. The analysis of Lemma 3.2 in [10] then holds The analysis in Lemma 2 then holds with replacing "Steps 1 and 4" in Algorithm 1 by "ALG 1," "Steps 2 and 5" in Algorithm 1 by "ALG 2," and "Steps 3 and 6" in Algorithm 1 by "ALG 3" where appropriate.

Next, we analyze the local convergence rate of the proposed algorithms. Assume that $\{x_k\}$ and $\{\lambda_k\}$ generated by the class of algorithms converge to local solution x^* and λ^*, respectively. In addition, they are contained in a convex set Ω Furthermore, we give the following assumptions.

3.1. Assumptions L

L1. The functions f and c are twice continuously differentiable in a neighborhood Ω of x^*.

L2. x^* and λ^* satisfy $\nabla_x \mathscr{L}(x^*, \lambda^*) = 0$.

L3. $A(x^*)$ is full column rank.

L4. There exists a constant $\kappa > 0$ such that $u^T W_k u \ge \kappa \|u\|^2$ for $A_k^T u = 0$ and W^* is positive definite on the null space of $A(x^*)^T$. The sequence $\{W_k\}$ is uniformly bounded.

L5. W_k satisfies that

$$\lim_{k \longrightarrow \infty} \frac{\left\| \left[W_k - \nabla_{xx}^2 \mathscr{L}(x_k, \lambda_{k+1}) \right] d_k \right\|}{\|d_k\|} = 0. \tag{3}$$

For the sake of local convergence analysis, we define a penalty function

$$\phi_\omega(x,\lambda) := f(x) + \lambda^T c(x) + \omega h(x). \tag{4}$$

The local linearized approximation of the penalty function is

$$q_\omega\left(x_k, d, \lambda_k, d^\lambda\right) = f_k + g_k^T d + \left(\lambda_k + d^\lambda\right)^T \left(c_k + A_k^T d\right) + \frac{1}{2}u^T W_k u + \frac{\omega}{2}\left\|c_k + A_k^T d\right\|^2. \tag{5}$$

Lemma 1. *If Assumptions L hold and $\eta_k = O(\|d_k\|)$, then there exists a constant $\widetilde{\gamma}_\omega > 0$ such that when $\omega \geq \widetilde{\gamma}_\omega$,*

$$q_\omega(x_k, 0, \lambda_k, 0) - q_\omega\left(x_k, d_k, \lambda_k, d_k^\lambda\right) \geq 0. \tag{6}$$

Proof. From (2d) and the assumption of global convergence, we can obtain that $v_k = O(\|c_k\|)$. (5) implies that

$$q_\omega(x_k, 0, \lambda_k, 0) - q_\omega\left(x_k, d_k, \lambda_k, d_k^\lambda\right) = \frac{\omega}{2}\left\|c_k\right\|^2 + \lambda_k^T c_k - g_k^T d_k - \frac{1}{2}u_k^T W_k u_k. \tag{7}$$

For ALG 1, we have that

$$\begin{aligned} u_k^T W_k u_k &= u_k^T\left[-g_k + A_k\left(A_k^T W_k^{-1} A_k\right)^{-1} A_k^T W_k^{-1} g_k + r_k\right] \\ &= -u_k^T g_k + r_k^T W_k^{-1} A_k\left(A_k^T W_k^{-1} A_k\right)^{-1} A_k^T W_k^{-1} g_k + u_k^T r_k, \end{aligned} \tag{8}$$

and

$$g_k^T u_k = -u_k^T W_k u_k + g_k^T W_k^{-1} A_k\left(A_k^T W_k^{-1} A_k\right)^{-1} A_k^T W_k^{-1} r_k + r_k^T u_k. \tag{9}$$

Since $\eta_k = O(\|d_k\|)$, we have $\|r_k\| = O(\|u_k\|\|c_k\|) + O(\|c_k\|^2)$. Hence, from (2a)–(2h), (7), (9), and L4, we can get that

$$\begin{aligned} q_\omega(x_k, 0, \lambda_k, 0) - q_\omega\left(x_k, d_k, \lambda_k, d_k^\lambda\right) &= \frac{\omega}{2}\left\|c_k\right\|^2 + \frac{1}{2}u_k^T W_k u_k + O\left(\|u_k\|\|c_k\|\right) + O\left(\|c_k\|^2\right) \\ &\geq \left(\frac{\omega}{2} - \kappa_3\right)\|c_k\|^2 - \kappa_4\|u_k\|\|c_k\| + \frac{1}{2}\kappa\|u_k\|^2, \end{aligned} \tag{10}$$

where $\kappa_3, \kappa_4 > 0$. It follows with $\theta_k := \|u_k\|/\|c_k\|$ that

$$\frac{1}{\|c_k\|^2}\left[q_\omega(x_k, 0, \lambda_k, 0) - q_\omega\left(x_k, d_k, \lambda_k, d_k^\lambda\right)\right] \geq \left(\frac{\omega}{2} - \kappa_3\right) - \kappa_4\theta_k + \frac{1}{2}\kappa\theta_k^2 = \frac{1}{2}\kappa\left(\theta_k - \frac{\kappa_4}{\kappa}\right)^2 + \frac{\omega}{2} - \kappa_3 - \frac{\kappa_4^2}{2\kappa}. \tag{11}$$

Let $\widetilde{\gamma}_\omega := 2\kappa_3 + \kappa_4^2/\kappa$. Then, (6) holds for $\omega \geq \widetilde{\gamma}_\omega$.
For ALG 2, we obtain that

$$\begin{aligned} u_k &= P_k\widehat{u}_k = -W_k^{-1} g_k + W_k^{-1} A_k\left(A_k^T W_k^{-1} A_k\right)^{-1} A_k^T W_k^{-1} g_k + P_k W_k^{-1} r_k, \\ u_k^T W_k u_k &= u_k^T\left(-\nabla_x \mathscr{L} + r_k\right) - u_k^T A_k\left(A_k^T W_k^{-1} A_k\right)^{-1} A_k^T W_k^{-1}\left(-\nabla_x \mathscr{L} + r_k\right) \\ &\quad - u_k^T g_k + u_k^T r_k + u_k^T A_k\left(A_k^T W_k^{-1} A_k\right)^{-1} A_k^T W_k^{-1} g_k - u_k^T A_k\left(A_k^T W_k^{-1} A_k\right)^{-1} A_k^T W_k^{-1} r_k. \end{aligned} \tag{12}$$

Hence,

$$g_k^T u_k = -u_k^T W_k u_k + r_k^T u_k - r_k^T W_k^{-1} A_k \left(A_k^T W_k^{-1} A_k\right)^{-1} A_k^T u_k. \tag{13}$$

Using (7), (13), and L4, we have that

$$\begin{aligned} q_\omega(x_k, 0, \lambda_k, 0) - q_\omega\left(x_k, d_k, \lambda_k, d_k^\lambda\right) &= \frac{\omega}{2}\|c_k\|^2 + \frac{1}{2}u_k^T W_k u_k + \lambda_k^T c_k - r_k^T u_k \\ &+ r_k^T W_k^{-1} A_k \left(A_k^T W_k^{-1} A_k\right)^{-1} A_k^T u_k - g_k^T v_k \\ &= \frac{\omega}{2}\|c_k\|^2 + \frac{1}{2}u_k^T W_k u_k + O\left(\|u_k\|\|c_k\|\right) + O\left(\|c_k\|^2\right) \\ &\geq \left(\frac{\omega}{2} - \widehat{\kappa}_3\right)\|c_k\|^2 - \widehat{\kappa}_4\|u_k\|\|c_k\| + \frac{1}{2}\kappa\|u_k\|^2, \end{aligned} \tag{14}$$

where $\widehat{\kappa}_3, \widehat{\kappa}_4 > 0$. Define $\theta_k := \|u_k\|/\|c_k\|$, and (14) then implies that

$$\frac{1}{\|c_k\|^2}\left[q_\omega(x_k, 0, \lambda_k, 0) - q_\omega\left(x_k, d_k, \lambda_k, d_k^\lambda\right)\right] \geq \frac{1}{2}\kappa\left(\theta_k - \frac{\widehat{\kappa}_4}{\kappa}\right)^2 + \frac{\omega}{2} - \widehat{\kappa}_3 - \frac{\widehat{\kappa}_4^2}{2\kappa}. \tag{15}$$

Let $\widetilde{\gamma}_\omega := 2\widehat{\kappa}_3 + \widehat{\kappa}_4^2/\kappa$. Then, (6) holds for $\omega \geq \widetilde{\gamma}_\omega$. For ALG 3, (2a)–(2h) imply that

$$\begin{aligned} u_k &= \left(I - A_k\left[A_k^T A_k\right]^{-1} A_k^T\right)\widehat{u}_k \\ &= -W_k^{-1} g_k + W_k^{-1} A_k \left(A_k^T W_k^{-1} A_k\right)^{-1} A_k^T W_k^{-1} g_k - W_k^{-1} A_k \left(A_k^T W_k^{-1} A_k\right)^{-1} c_k + A_k \left(A_k^T A_k\right)^{-1} c_k + P_k W_k^{-1} r_k \\ &= \widehat{u}_k - W_k^{-1} r_k - v_k + P_k W_k^{-1} r_k, \\ u_k^T W_k u_k &= u_k^T W_k \widehat{u}_k - u_k^T r_k - u_k^T W_k v_k + u_k^T W_k P_k W_k^{-1} r_k = -u_k^T g_k - u_k^T W_k v_k + u_k^T W_k P_k W_k^{-1} r_k, \end{aligned} \tag{16}$$

and

$$g_k^T u_k = -u_k^T W_k u_k - u_k^T W_k v_k + u_k^T W_k P_k W_k^{-1} r_k. \tag{17}$$

From (7), (17), and L4, we deduce that

$$\begin{aligned} q_\omega(x_k, 0, \lambda_k, 0) - q_\omega\left(x_k, d_k, \lambda_k, d_k^\lambda\right) &= \frac{\omega}{2}\|c_k\|^2 + \lambda_k^T c_k + \frac{1}{2}u_k^T W_k u_k + u_k^T W_k v_k \\ &- u_k^T W_k P_k W_k^{-1} r_k - g_k^T v_k \\ &= \frac{\omega}{2}\|c_k\|^2 + \frac{1}{2}u_k^T W_k u_k \\ &+ O\left(\|u_k\|\|c_k\|\right) + O\left(\|c_k\|^2\right). \end{aligned} \tag{18}$$

Similarly, we can also prove that there exists $\widetilde{\gamma}_\omega$ such that (6) holds for $\omega \geq \widetilde{\gamma}_\omega$. □

Lemma 2. *Suppose Assumptions L hold and $\eta_k = O(\|d_k\|)$. There exists a neighborhood $\mathscr{B}_1$ of x^* such that if the first equation in Step 6 in Algorithm 1 holds for $\alpha_{k,l} = 1$, then the trial step is accepted by the Armijo condition (the second equation in Step 6 in Algorithm 1) for $x_k \in \mathscr{B}_1$.*

Proof. If the first equation in Step 6 in Algorithm 1 holds, $s_{\mathscr{L}} > 2s_h$ implies that

$$\begin{aligned} \frac{1}{2}\|c_k\|^2 = h(x_k) &< \tau_1^{-1/s_h}\left(-m_k(1)\right)^{s_{\mathscr{L}}/s_h} \\ &< \tau_1^{-1/s_h}\left(-g_k^T d_k + \lambda_k^T c_k\right)^{s_{\mathscr{L}}/s_h} \\ &= \tau_1^{-1/s_h}\left(-g_k^T u_k + O\left(\|v_k\|^2\right)\right)^{s_{\mathscr{L}}/s_h} \\ &= o\left(\|u_k\|^2\right). \end{aligned} \tag{19}$$

By (9), (13), (17), and $r_k, v_k = O(\|c_k\|)$, we can easily get

(1) Initialize: choose starting point x_0, λ_0, and $W_0 = I$. Set constants $\beta \in (0, 1/2)$, $\gamma_h, \gamma_{\mathscr{L}} \in (0,1)$, $t \in (0, 1-\gamma_h/1+\gamma_h)$, $\tau_1, \nu > 0$, $s_h > 1$, $s_{\mathscr{L}} > 2s_h$, $\gamma_\alpha \in (0,1]$, $\varsigma \in (0,1)$, and $\varepsilon > 0$. Choose an initial filter $\mathscr{F}_0 := \{(h, \mathscr{L}) \in \mathbb{R}^2 : h \geq \nu\}$ and let $k := 0$.
(2) If $\left\| \begin{pmatrix} \nabla_x \mathscr{L}(x_k, \lambda_k) \\ c_k \end{pmatrix} \right\| < \varepsilon$, then stop.
(3) Evaluate f_k, g_k, c_k, and A_k. Then, compute the multiplier $\lambda_{k+1} = U(x_k, \lambda_k, W_k)$ from (2a). Find an inexact step d_k from (2b)–(2e). Compute $d_k^\lambda = \lambda_{k+1} - \lambda_k$. If this system ((2a)–(2h)) is too ill-conditioned, go to Step 9.
(4) Set $\alpha_{k,0} = 1$ and $l = 0$.
(5) If $\alpha_{k,l} < \alpha_k^{\min}$ with
$$\alpha_k^{\min} := \gamma_\alpha \cdot \begin{cases} \min\left\{\gamma_h, \left(\gamma_{\mathscr{L}} h_k / \left(-g_k^T d_k + \lambda_k^T c_k - |(d_k^\lambda)^T c_k|\right)\right), \left(\tau_1 h_k^{s_h} / \left(\left(-g_k^T d_k + \lambda_k^T c_k - |(d_k^\lambda)^T c_k|\right)^{s_{\mathscr{L}}}\right)\right)\right\} & \text{if } g_k^T d_k - \lambda_k^T c_k + |(d_k^\lambda)^T c_k| < 0 \\ \gamma_h, & \text{otherwise} \end{cases}$$
go to Step 9. Otherwise, if $(h(x_k + \alpha_{k,l} d_k), \mathscr{L}(x_k + \alpha_{k,l} d_k, \lambda_k + \alpha_{k,l} d_k^\lambda)) \in \mathscr{F}_k$, then go to Step 7.
(6) Define $m_k(\alpha) := \alpha g_k^T d_k - \alpha \lambda_k^T c_k + \alpha(1-\alpha) c_k^T d_k^\lambda$. If
$$m_k(\alpha_{k,l}) < 0, \ [-m_k(\alpha_{k,l})]^{s_{\mathscr{L}}} \alpha_{k,l}^{1-s_{\mathscr{L}}} > \tau_1 h(x_k)^{s_h}$$
and
$$\mathscr{L}(x_k + \alpha_{k,l} d_k, \lambda_k + \alpha_{k,l} d_k^\lambda) \leq \mathscr{L}(x_k, \lambda_k) + \beta m_k(\alpha_{k,l})$$
hold, set $x_{k+1} := x_k + \alpha_{k,l} d_k$ and go to Step 8. If the first equation in Step 6 in Algorithm 1 does not hold, then if
$$h(x_k + \alpha_{k,l} d_k) \leq (1 - \gamma_h) h(x_k)$$
or
$$\mathscr{L}(x_k + \alpha_{k,l} d_k, \lambda_{k+1}) \leq \mathscr{L}(x_k, \lambda_k) - \gamma_{\mathscr{L}} h(x_k)$$
holds, set $x_{k+1} := x_k + \alpha_{k,l} d_k$ and go to Step 8.
(7) Set $\alpha_{k,l+1} = \varsigma \alpha_{k,l}$ and $l \leftarrow l + 1$. Go to Step 5.
(8) Set $\alpha_k := \alpha_{k,l}$. If the first equation in Step 6 in Algorithm 1 does not hold for α_k, augment the filter by
$$\mathscr{F}_{k+1} = \mathscr{F}_k \cup \{(h, \mathscr{L}) \in \mathbb{R}^2 : h \geq (1-\gamma_h) h(x_k) \text{ and } \mathscr{L} \geq \mathscr{L}(x_k, \lambda_k) - \gamma_{\mathscr{L}} h(x_k)\}$$
otherwise, set $\mathscr{F}_{k+1} := \mathscr{F}_k$. Go to Step 10.
(9) Find a new point x_{k+1} that satisfies the following conditions:
  (a) $h(x_k)$ is greater than $h(x_{k+1})$
  (b) The third and fourth equations in Step 6 in Algorithm 1 hold for x_{k+1}
  (c) $(h(x_{k+1}), \mathscr{L}(x_{k+1}, \lambda_{k+1})) \notin \mathscr{F}_k$

  Augment the filter by the equation in Step 8 in Algorithm 1. Go to Step 10.
(10) Compute $y_k = P_k[\nabla_x \mathscr{L}(x_k + u_k, \lambda_k) - \nabla_x \mathscr{L}(x_k, \lambda_k)]$ and update W_k by (2g). Set $k := k + 1$ and go to Step 2.

Algorithm 1: ISLSF.

$$\begin{aligned} g_k^T d_k &= -u_k^T W_k u_k + O\left(\|u_k\| \|c_k\|\right) + O\left(\|r_k\|\right) \\ &= -u_k^T W_k u_k + o\left(\|u_k\|^2\right). \end{aligned} \tag{20}$$

Hence, by Assumption L5, (19), and (20), we see that

$$\begin{aligned} &\mathscr{L}(x_k, \lambda_k) - \mathscr{L}\left(x_k + d_k, \lambda_k + d_k^\lambda\right) + \beta\left(g_k^T d_k - \lambda_k^T c_k\right) \\ &\quad = \lambda_k^T c_k - g_k^T d_k - \frac{1}{2} d_k^T \nabla^2 f(x_k) d_k - \left(\lambda_k + d_k^\lambda\right)^T c(x_k + d_k) + \beta\left(g_k^T d_k - \lambda_k^T c_k\right) + o\left(\|d_k\|^2\right) \\ &\quad = (\beta - 1) g_k^T d_k + (1-\beta) \lambda_k^T c_k - \frac{1}{2} d_k^T \nabla^2 f(x_k) d_k - \frac{1}{2} d_k^T \left[\sum_{i=1}^m \lambda_{k,i} \nabla^2 c_i(x_k)\right] d_k + o\left(\|d_k\|^2\right) \\ &\quad = (1-\beta) u_k^T W_k u_k - \frac{1}{2}\left(u_k^T W_k u_k + 2 u_k^T W_k v_k + v_k^T W_k v_k\right) + o\left(\|u_k\|\right) + o\left(\|d_k\|^2\right) \\ &\quad = \left(\frac{1}{2} - \beta\right) u_k^T W_k u_k + O\left(\|c_k\|\right) + o\left(\|u_k\|^2\right) = \left(\frac{1}{2} - \beta\right) u_k^T W_k u_k + o\left(\|u_k\|^2\right). \end{aligned} \tag{21}$$

From Assumption L4, $\beta \in (0, 1/2)$, and we can conclude that the Armijo condition (the second equation in Step 6 in Algorithm 1) holds for k sufficiently large. □

Next, we present a result which is necessary for the local convergence.

Lemma 3. *Suppose Assumptions L hold. Choose $\eta_k = O(\|d_k\|)$. There exist a neighborhood $\mathscr{B}_2 \subseteq \mathscr{B}_1$ and a constant $\gamma_\omega > 0$ such that for $\omega \geq \gamma_\omega$ and $x_k \in \mathscr{B}_2$,*

$$\phi_\omega(x_k, \lambda_k) - \phi_\omega\left(x_k + d_k, \lambda_k + d_k^\lambda\right) \geq \frac{1+\gamma_h}{2}\left[q_\omega(x_k, 0, \lambda_k, 0) - q_\omega\left(x_k, d_k, \lambda_k, d_k^\lambda\right)\right]. \tag{22}$$

Proof. From (20) and $v_k = O(\|c_k\|)$, we have that

$$\begin{aligned}
&\phi_\omega(x_k, \lambda_k) - \phi_\omega\left(x_k + d_k, \lambda_k + d_k^\lambda\right) \\
&\quad = f_k + \lambda_k^T c_k + \frac{\omega}{2}\|c_k\|^2 - f(x_k + d_k) - \left(\lambda_k + d_k^\lambda\right)^T c(x_k + d_k) - \frac{\omega}{2}\|c(x_k + d_k)\|^2 \\
&\quad = \lambda_k^T c_k + \frac{\omega}{2}\|c_k\|^2 - g_k^T d_k - \frac{1}{2} d_k^T \nabla^2 f(x_k) d_k - \lambda_k^T c(x_k + d_k) + o\left(\|d_k\|^2\right) \\
&\quad = \frac{\omega}{2}\|c_k\|^2 - g_k^T d_k + \lambda_k^T c_k - \frac{1}{2} d_k^T \nabla^2 f(x_k) d_k - \frac{1}{2} d_k^T \left[\sum_{i=1}^m \lambda_{k,i} \nabla^2 c_i(x_k)\right] d_k + o\left(\|d_k\|^2\right) \\
&\quad = \frac{\omega}{2}\|c_k\|^2 + u_k^T W_k u_k - \frac{1}{2} d_k^T W_k d_k + O\left(\|u_k\|\|c_k\|\right) + O\left(\|c_k\|^2\right) + o\left(\|d_k\|^2\right) \\
&\quad = \frac{\omega}{2}\|c_k\|^2 + u_k^T W_k u_k - \frac{1}{2}\left(u_k^T W_k u_k + 2u_k^T W_k v_k + v_k^T W_k v_k\right) + O\left(\|u_k\|\|c_k\|\right) + O\left(\|c_k\|^2\right) + o\left(\|d_k\|^2\right) \\
&\quad = \frac{\omega}{2}\|c_k\|^2 + \frac{1}{2} u_k^T W_k u_k + O\left(\|u_k\|\|c_k\|\right) + O\left(\|c_k\|^2\right) + o\left(\|u_k\|^2\right).
\end{aligned} \tag{23}$$

Hence, from Assumption L2, (10), (14), (18), (23), and $\gamma_h \in (0, 1)$, we obtain that

$$\begin{aligned}
&\phi_\omega(x_k, \lambda_k) - \phi_\omega\left(x_k + d_k, \lambda_k + d_k^\lambda\right) - \frac{1+\gamma_h}{2}\left[q_\omega(x_k, 0, \lambda_k, 0) - q_\omega\left(x_k, d_k, \lambda_k, d_k^\lambda\right)\right] \\
&\quad = \frac{\omega}{2}\|c_k\|^2 + \frac{1}{2} u_k^T W_k u_k + O\left(\|u_k\|\|c_k\|\right) + O\left(\|c_k\|^2\right) + o\left(\|u_k\|^2\right) - \frac{1+\gamma_h}{2}\left[\frac{\omega}{2}\|c_k\|^2 + \frac{1}{2} u_k^T W_k u_k + O\left(\|u_k\|\|c_k\|\right) + O\left(\|c_k\|^2\right)\right] \\
&\quad = \frac{1-\gamma_h}{4}\omega\|c_k\|^2 + \frac{1-\gamma_h}{4} u_k^T W_k u_k + O\left(\|u_k\|\|c_k\|\right) + O\left(\|c_k\|^2\right) + o\left(\|u_k\|^2\right) \\
&\quad \geq \frac{1-\gamma_h}{4}\kappa\|u_k\|^2 - \kappa_5\|u_k\|\|c_k\| + \left(\frac{1-\gamma_h}{4}\omega - \kappa_6\right)\|c_k\|^2,
\end{aligned} \tag{24}$$

where $\kappa_5, \kappa_6 > 0$. Set $\theta_k := \|u_k\|/\|c_k\|$. Therefore, we can get

$$\begin{aligned}
&\frac{1}{\|c_k\|^2}\left\{\phi_\omega(x_k,\lambda_k)-\phi_\omega\left(x_k+d_k,\lambda_k+d_k^\lambda\right)-\frac{1+\gamma_h}{2}\right.\\
&\qquad\left.\left[q_\omega(x_k,0,\lambda_k,0)-q_\omega\left(x_k,d_k,\lambda_k,d_k^\lambda\right)\right]\right\}\\
&\geq\frac{1-\gamma_h}{4}\kappa\theta_k^2-\kappa_5\theta_k+\left(\frac{1-\gamma_h}{4}\omega-\kappa_6\right)\\
&=\frac{1-\gamma_h}{4}\kappa\left[\theta_k-\frac{2\kappa_5}{(1-\gamma_h)\kappa}\right]^2+\frac{1-\gamma_h}{4}\omega-\kappa_6\\
&\quad-\frac{\kappa_5^2}{(1-\gamma_h)\kappa}.
\end{aligned}\tag{25}$$

Define $\gamma_\omega := 4\kappa_6/1-\gamma_h+4\kappa_5^2/(1-\gamma_h)^2\kappa$. We conclude that (22) holds for $\omega\geq\gamma_\omega$. □

From [7], we can easily prove that Lemma 3 is true for the next three choices of steps.

$$\overline{d}_k=d_k+d_{k+1},\ \overline{d}_k^\lambda=d_k^\lambda+d_{k+1}^\lambda,\tag{26a}$$

$$\overline{d}_k=d_k+d_{k+1}+d_{k+2},\ \overline{d}_k^\lambda=d_k^\lambda+d_{k+1}^\lambda+d_{k+2}^\lambda,\tag{26b}$$

$$\text{or }\overline{d}_k=d_k+d_{k+1}+d_{k+2}+d_{k+3},\ \overline{d}_k^\lambda=d_k^\lambda+d_{k+1}^\lambda+d_{k+2}^\lambda+d_{k+3}^\lambda.\tag{26c}$$

Using arguments similar to those in the proof of Lemma 4.5, Lemma 4.6, and Theorem 4.7 in [7], we are able to show the main local convergence theorem.

Theorem 1. *Suppose Assumptions L hold. For k sufficiently large, the sequence $\{x_k\}$ converges to x^* with a superlinear rate.*

4. Numerical Results

We have performed numerical experiments on a personal computer with Intel CORE i5-6200U/4GB/1TB. The numerical results have been obtained by running our algorithms on the set of 43 equality constrained problems. These problems were taken from the following web page: https://www.gamsworld.org/performance/princetonlib/htm/group5stat.htm.

In Table 1, Name, N, and M stand for the name of the chosen test problem, the number of variable, and the number of constraint, respectively.

We used the starting point supplied with these problems, wrote all codes in the Matlab code, and limited all attempts to solve these test problems to a maximum of 1000 iterations. The parameters in these algorithms were chosen as follows: $\beta=0.3$, $\gamma_h=0.01$, $\gamma_{\mathscr{L}}=0.5$, $\tau_1=1$, $\nu=10^4$, $s_h=1.5$, $s_{\mathscr{L}}=3.2$, $\gamma_\alpha=0.5$, and $\varsigma=0.5$, which seemed to work reasonably well for a broad class of problems. Update the Hessian matrix approximation W_k by the BFGS secant update formula

Table 1: Numerical comparisons.

Name	N	M
eigena2	110	55
eigencco	30	14
fccu	19	8
hs007	2	1
hs039	4	2
hs077	6	2
hs08	2	2
hs046	5	2
hs050	5	2
hs056	7	4
bt2	3	1
bt6	5	2
hs100lnp	7	2
bt12	5	3
mwright	5	3
eigenaco	110	55
hatfldg	25	24
genhs28	10	8
hs027	3	1
hs040	4	3
hs078	5	3
hs026	3	1
hs048	5	2
hs051	5	3
hs061	3	2
bt3	5	3
bt7	5	3
bt10	2	2
bt5	3	2
eigenbco	110	55
robot	14	9
hs111lnp	10	3
hs028	3	1
hs047	5	3
hs079	5	3
hs042	4	2
hs049	5	2
hs052	5	3
hs009	2	1
bt4	3	2
bt9	4	2
bt11	5	3
maratos	2	1

$$W_{k+1}=B(u_k,y_k,W_k)=W_k+\frac{y_ky_k^T}{y_k^Tu_k}-\frac{(W_ku_k)(W_ku_k)^T}{u_k^TW_ku_k}.\tag{27}$$

The computation terminated if the following stopping criterion was satisfied:

$$\left\|\begin{pmatrix}\nabla_x\mathscr{L}(x_k,\lambda_k)\\ c_k\end{pmatrix}\right\|\leq 10^{-5}.\tag{28}$$

We summarize the numerical results in Figures 1(a)–1(d) in terms of the performance profiles as outlined in [11]. Our profiles are based on the number of iterations, that of evaluating function, that of augment of the filter set, and the CPU times (seconds). The value of $\rho(\tau)$ for $\tau=1$ represents

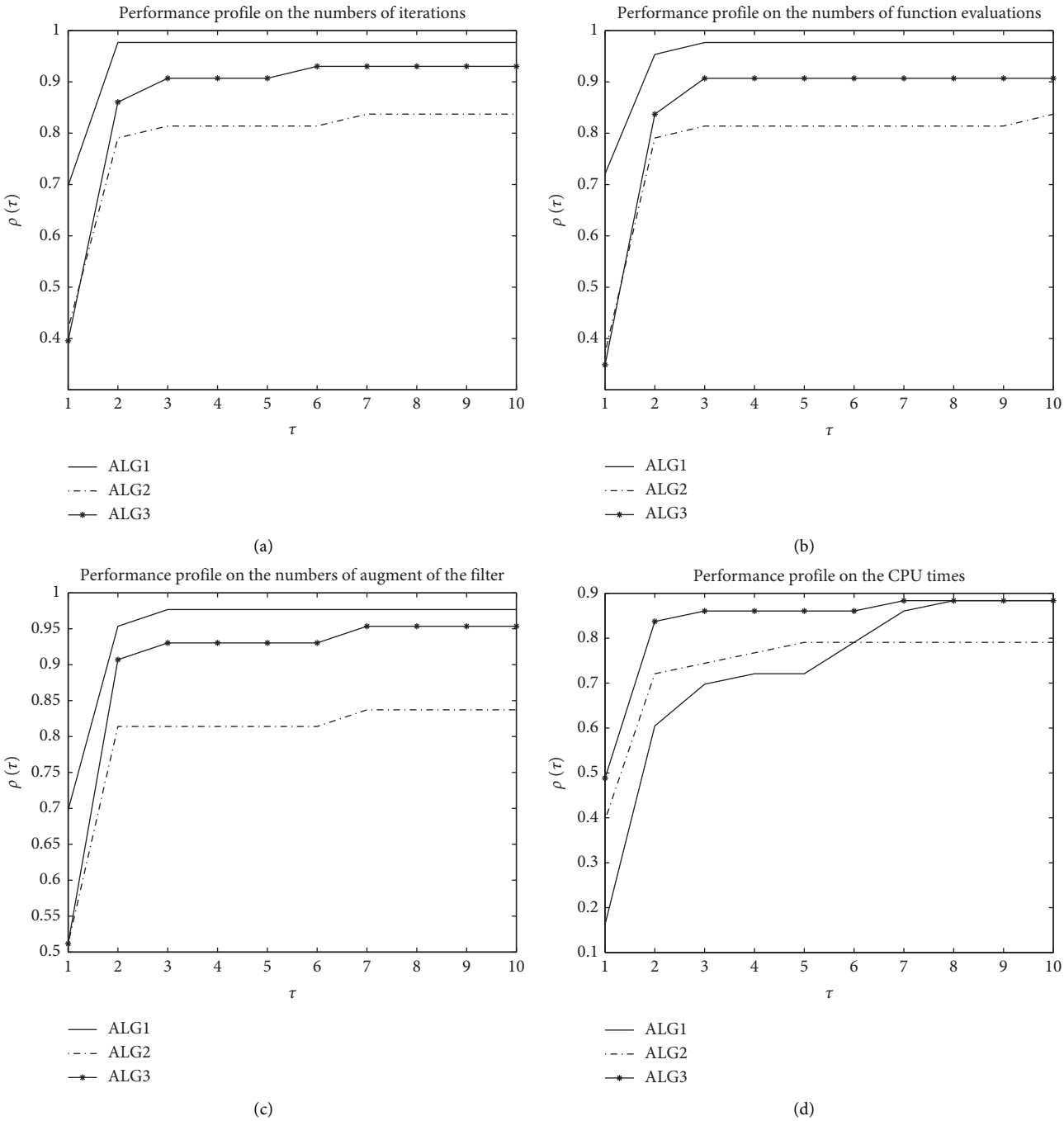


Figure 1: Performance profile on [1, 10].

the probability that the algorithm will win over the others. Function values $\rho(\tau)$ on the rightmost values illustrate the robustness of the approach.

Figures 1(a) and 1(b) show the performance profiles for the number of iterations and that of evaluating function required to solve the problems, by the three different algorithms. We can observe from the two figures that ALG 1, which solves 98% of the problems, is superior to the next two. Because of solving 83% of those, ALG 2 is inferior to ALG 3, which solves 92% of those. From this figure, it is clear that ALG 1 has the most wins (has the highest probability of being the optimal algorithm) and that the probability that ALG 1 is the winner on a given problem is about 0.7.

As is known to all, the fewer number the filter set augments, the better performance the algorithm achieves. From Figure 1(c), we may analyze that ALG 1 is the most efficient on almost 70% of the problems when the number of the filter expansions is under comparison. Observing the right side of the plots in the figure, we conclude that ALG 1 solves most problems to optimality (approximately 97.5%).

A CPU-time performance profile is presented in Figure 1(d) for the given test problems and these 3 algorithms. An observation that emerges from the figure is that ALG 3 clearly outperforms the other algorithms. The second conclusion is that ALG 1 is by far the worst with $\tau \leq 6$. Another point of interest is that ALG 1 and ALG 3 have the best probability $\rho(\tau)$ for τ in the interval $\tau \geq 8$, with similar performance.

These performance profiles can indicate that the proposed three algorithms are effective and robust.

5. Conclusions

In this paper, we propose a class of inexact secant line search filter methods for solving nonlinear equality constrained optimization. The local convergence properties and numerical performance of the class of algorithms are presented. A simple modification of these methods proposed in a companion paper [7] avoiding second-order correction steps is given. Our methods do not suffer from the Maratos effect by using Lagrangian function $\mathscr{L}$ as the objective function and $\|c(x)\|^2$ as the constraint violation. We compare these three methods, and detailed numerical results indicate that the proposed algorithms are robust and effective, while ALG 1 is superior to the next two.

Data Availability

The data supporting the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported by Scientific Research Fund of Hunan Provincial Education Department (no. 18A351).

References

[1] F. Arzani and M. R. Peyghami, "An approach based on dwindling filter method for positive definite generalized eigenvalue problem," *Computational and Applied Mathematics*, vol. 37, no. 2, pp. 1197–1212, 2018.

[2] R. H. Byrd, F. E. Curtis, and J. Nocedal, "An inexact SQP method for equality constrained optimization," *SIAM Journal on Optimization*, vol. 19, no. 1, pp. 351–369, 2008.

[3] G. Eichfelder, K. Klamroth, and J. Niebling, "Nonconvex constrained optimization by a filtering branch and bound," *Journal of Global Optimization*, vol. 80, no. 1, pp. 31–61, 2021.

[4] C. Gu, D. Zhu, and Y. Pei, "A new inexact SQP algorithm for nonlinear systems of mixed equalities and inequalities," *Numerical Algorithms*, vol. 78, no. 4, pp. 1233–1253, 2018.

[5] D. Li and D. Zhu, "An affine scaling interior trust-region method combining with line search filter technique for optimization subject to bounds on variables," *Numerical Algorithms*, vol. 77, no. 4, pp. 1159–1182, 2018.

[6] A. Wächter and L. T. Biegler, "Line search filter methods for nonlinear programming: motivation and Global convergence," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 1–31, 2005.

[7] A. Wächter and L. T. Biegler, "Line search filter methods for nonlinear programming: local convergence," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 32–48, 2005.

[8] R. Fontecilla, "Local convergence of secant methods for nonlinear constrained optimization," *SIAM Journal on Numerical Analysis*, vol. 25, no. 3, pp. 692–712, 1988.

[9] R. S. Dembo, S. C. Eisenstat, and T. Steihaug, "Inexact Newton methods," *SIAM Journal on Numerical Analysis*, vol. 19, no. 2, pp. 400–408, 1982.

[10] Z. Wang, L. Cai, and D. Zhu, "Line search filter inexact secant methods for nonlinear equality constrained optimization," *Applied Mathematics and Computation*, vol. 263, no. 2, pp. 47–58, 2015.

[11] E. D. Dolan and J. J. Moré, "Benchmarking optimization software with performance profiles," *Mathematical Programming*, vol. 91, no. 2, pp. 201–213, 2002.

Hindawi
Mathematical Problems in Engineering
Volume 2021, Article ID 6756588, 16 pages
https://doi.org/10.1155/2021/6756588

Research Article

Hyperheuristic Based Migrating Birds Optimization Algorithm for a Fairness Oriented Shift Scheduling Problem

Gözde Alp and Ali Fuat Alkaya

Department of Computer Engineering, Engineering Faculty, Marmara University, 34722 Istanbul, Turkey

Correspondence should be addressed to Gözde Alp; gozde.alp@marmara.edu.tr

Received 14 September 2021; Revised 12 October 2021; Accepted 18 October 2021; Published 2 November 2021

Academic Editor: Guoqiang Wang

The purpose of this paper is twofold. First, it introduces a new hybrid computational intelligence algorithm to the optimization community. This novel hybrid algorithm has hyperheuristic (HH) neighborhood search movements embedded into a recently introduced migrating birds optimization (MBO) algorithm. Therefore, it is called HHMBO. Second, it gives the necessary mathematical model for a shift scheduling problem of a manufacturing company defined by including the fairness perspective, which is typically ignored especially in manufacturing industry. Therefore, we call this complex optimization problem fairness oriented integrated shift scheduling problem (FOSSP). HHMBO is applied on FOSSP and is compared with the well-known simulated annealing, hyperheuristics, and classical MBO algorithms through extended computational experiments on several synthetic datasets. Experiments demonstrate that the new hybrid computational intelligence algorithm is promising especially for large sized instances of the specific problem defined here. HHMBO has a high exploration capability and is a promising technique for all optimization problems. To justify this assertion, we applied HHMBO to the well-known quadratic assignment problem (QAP) instances from the QAPLIB. HHMBO was up to 14.6% better than MBO on converging to the best known solutions for QAP benchmark instances with different densities. We believe that the novel hybrid method and the fairness oriented model presented in this study will give new insights to the decision-makers in the industry as well as to the researchers from several disciplines.

1. Introduction

Workforce scheduling has been a subject of continued research and commercial interest in several disciplines due to its important practical applications within the context of intelligent systems. Workforce scheduling is a concept that embraces a variety of scheduling problems, also referred to as manpower scheduling and personnel scheduling in the literature. Constructing efficient and equitable schedules is a challenging issue requiring time and labor cost for the companies with high number of employees.

The real world problem we tackled aims to provide fair personnel work schedules for a large-scale manufacturing company that works 24 hours and seven days a week. Personnel requirements for each day and shift differ and are updated periodically. Employee requirement changes every four weeks, regularly. On the other hand, there is a high rate of employee circulation. Large numbers of new employees are recruited and large numbers of employees leave at the end of each period. Therefore, possible maximum level of fairness within each planning period is sought. There are also some essential legal regulations for employee schedules which make the problem much more complicated. The problem is described in detail in Section 3.

The presented problem has a sophisticated solution space. We have envisioned that exploring the solution space by more than one neighborhood search method could contribute to finding promising results. Hyperheuristics is a widely known way of applying multiple heuristic techniques. On the other hand, recently proposed migrating birds optimization algorithm is receiving growing attention from the researchers, due to the ability of providing good quality solutions over a wide range of instances of a problem. As a result, we wanted to integrate the exploitation capability of

MBO with HH's exploration capability for solving the problem. Consequently, we introduce a novel technique that has hyperheuristic movements embedded in the migrating birds optimization algorithm (HHMBO). To the best of our knowledge, there has been no investigation about hybridizing hyperheuristics with MBO in the literature. HHMBO is compared with the well-known algorithms, and it is shown that the proposed hybrid algorithm is promising especially for large sized instances of FOSSP.

HHMBO obtains very successful results on the FOSSP model. Hence, we believe HHMBO is also promising for the other combinatorial optimization problems. Therefore, we aimed to measure its performance on a widely tackled problem's benchmark instances. Based on this idea, we have implemented HHMBO on the quadratic assignment problem (QAP). As a result, HHMBO is again found to be very successful on converging to the best known solutions for QAP instances with different densities. We should note that the data as well as the source codes of the programming framework can be found at http://mimoza.marmara.edu.tr/falkaya/research.htm.

The contribution of this study is threefold: (i) defining and validating the mathematical model of the fairness oriented shift scheduling problem (FOSSP), (ii) proposing a new hybrid metaheuristic algorithm that is composed of hyperheuristic movements embedded in migrating birds optimization (HHMBO), and (iii) showing the superiority of HHMBO through computational experiments conducted on both FOSSP dataset and well-known QAP benchmark instances.

The remainder of the manuscript is organized as follows: Literature review is presented in Section 2, and problem description details are presented in Section 3. Proposed solution techniques are described in detail in Section 4. Experimental setup is demonstrated in Section 5. Results are presented and investigated in Section 6. Section 7 concludes the paper and comprises suggestions for further research.

2. Literature Review

Many scientific studies have been examined in order to locate the problem that we have dealt with. Ernst et al. [1] offered a guide study that consisted of an annotated bibliography of personnel scheduling. There are also many review studies that extensively examine the workforce scheduling literature [2–6]. Brucker et al. [7] categorized the shift scheduling problems. Baker classified workforce scheduling problems into three types [8]: shift scheduling problem, days-off scheduling problem, and tour scheduling problem. Shift scheduling involves selecting a set of the most suitable shifts from a (large) pool of candidate shifts on a single day by satisfying employee requirement in each time slot. On the other hand, the main concern in days-off scheduling is to determine the off-work days for each worker over the rostering horizon rather than to assign the worker particular shifts on working days. In contrast to days-off scheduling, tour scheduling chooses off days for the workers and decides shift types for each of their working days over the planning horizon. Mathematical model of tour scheduling problem is summarized and problem classification is presented by Alfares [9]. Erhard et al. [4] presented the first review study that focuses on quantitative methods for physician scheduling literature. They indicated in their study that it is more common to include fairness aspects for physician scheduling domain than to ignore them.

Many different workforce scheduling problem models are presented up to now. A new mathematical formulation for mine shift scheduling is presented by Seifi et al. [10]. Brunner and Stolletz [11] prepared a schedule for the employees in check-in counters at airport. Atlason et al. [12] presented a shift assignment study in which planning horizon is divided into short periods. Morris and Showalter [13] presented a rewarding study that clarifies the usage of set covering formulation for shift scheduling, days-off scheduling, and tour scheduling problems ([14–16]). Alfares [17] proposed a specific type of days-off scheduling, solved by proposed solution technique. Altner et al. [18] presented a days-off scheduling formulation. Lau [19] introduced a special tour scheduling problem.

The model we offer in this study associates tour scheduling problem and employee assignment problem and incarnates them as a single problem by overseeing the fairness issue. We call this integrated problem fairness oriented shift scheduling problem (FOSSP), in which schedules are constructed for employees during the planning period where the objective is providing fairness among all workers. The problem we are tackling is undertaken by solving the tour scheduling and employee assignment problems consecutively in [20]. However, obtaining solutions to subproblems gives suboptimal results. Therefore, an integrated solution approach is necessary. The mathematical model of FOSSP is built from scratch and then the model is implemented with a solver. However, the solver finds the optimum result but with an exponential growth in run time, due to the NP-Hard nature of the problems. Therefore, heuristics and/or metaheuristic approaches that provide reasonable solutions for the problem are needed.

Many metaheuristic based solution techniques have been applied for solving the workforce scheduling problems so far. They include artificial bee colony algorithm [21], particle swarm optimization [22], migrating birds optimization [23], genetic algorithm (GA) [24], an adaptive multiple crossover GA [25], and a modified differential evolution (DE) algorithm ([26–29]). A tabu-search hyperheuristics solution for nurse scheduling problem is presented in Burke et al.'s work [30]. Pan et al. [31] proposed a hybrid heuristic, combining tabu search and large neighborhood search techniques. Hernández-Leandro et al. [32] presented a Lagrangian relaxation and a metaheuristic for the multiactivity shift scheduling problem. A hybrid discrete water wave optimization algorithm is presented for scheduling problems [33].

Combinatorial optimization problems have a sophisticated solution space and a single improvement technique is insufficient to obtain promising results. More than one neighborhood search method with different characteristics may be promising to obtain more efficient exploration in the solution space. One widely known way of applying multiple heuristic techniques is hyperheuristics. Hyperheuristics

(HH) is a metaheuristic technique formally defined as a heuristic to select and manage a set of low level heuristics. Many HH variations for optimization and scheduling problems have been defined up to now, such as tensor based hyperheuristic for nurse scheduling [34], column based hyperheuristic for bus driver scheduling problem [35], tabu-search hyperheuristics [30], and a simulated annealing hyperheuristic [36]. Hyperheuristics are also integrated with swarm based computational intelligence techniques such as hybrid particle swarm optimization [37], ant colony optimization based hyperheuristic [38], and hyperheuristic based artificial bee colony algorithm [39]. On the other hand, Duman et al. [40] recently proposed a new metaheuristic algorithm named migrating birds optimization (MBO). They applied their algorithm to quadratic assignment problems (QAP) and proved its efficiency.

3. Problem Description

In this section, we try to define and clarify the proposed and tackled problem. Firstly, the properties and limitations of the problem are explained. Then, cost measurement method is described. Lastly, the mathematical model of FOSSP is presented. The terms time period, time slot, and working slot are used interchangeably throughout this manuscript.

3.1. Properties and Limitations of the Problem. The main motivation behind defining this new problem is obtaining a fair distribution of working slots (time periods) among fixed number of employees during the specified planning period. Production factory operates incessantly 24 hours a day, seven days a week. There are totally three shifts in each day; those are day shift, evening shift, and night shift. Employee requirement list contains weekly employee demand for each day of the week and each shift of a day. The planning horizon which is specified as four weeks in our case may consist of several weeks. Employee requirement list is updated by management department at the end of each period, and same the requirement list is valid for all weeks in planning horizon. Multiweek employee schedules are prepared supplying personnel demand. That is, the output schedule must satisfy the number of workers needed for each day and shift, besides ensuring a fair distribution between employees along the given weeks.

Additionally, there are some essential legal regulations for employee schedules. Some of those rules may be listed as follows; if an employee has night shift on Sunday, (s)he cannot be assigned to day shift on the following Monday; if an employee has weekend off, he/she cannot be off on adjacent Monday and an employee cannot work more than six consecutive days. These additional legal constraints increase the complexity of the problem. Under these constraints, the aim is to provide a fair schedule distribution among employees throughout the planning period. Employees are assigned to a specific stationary shift type during one week and each employee has two off days. However, for each worker, shift types and off days may change from one week to another.

3.2. Determination of Schema Costs. A tour or a schema may be defined by a seven-lettered string containing D, E, N, or X which mean day shift, evening shift, night shift, and off day, respectively, and each letter corresponds to the days of the week starting from Monday up to Sunday. An example schema string is "XXEEEEE" which means Monday and Tuesday are off and the week continues with five successive evening shifts. Thus, there occur exactly 63 different schemata with three shift types and two days off rules.

Three shifts in a day (namely, day, evening, and night) and seven days of the week make a total of 21 time slots, starting from Monday day shift and ending with Sunday night shift. In the factory, day shift is between 07:00 and 15:00, evening shift is between 15:00 and 23:00, and night shift is between 23:00 and 07:00. Each employee is assigned to one of those shifts each week; additionally each employee has two off days during a week.

Taking individual employee preferences into account while modeling the problem was not appropriate, as it would make the already complex problem unsolvable. Hereby, the preferability of time zones has been determined by a survey conducted with the employees. We had classified the off days according to their desirability. We have identified five off day types: (i) weekend off, (ii) Friday-Saturday off, (iii) Sunday-weekday off, (iv) consecutive weekday off, and (v) separate weekday off. For determining the quality of a schema, we assigned numerical values to off day types and shift types. Based on the survey results obtained, enumerations for shift types are determined as one, two, and three for day, evening, and night shifts, respectively. Similarly, off day types, weekend off, Friday-Saturday off, Sunday-weekday off, consecutive weekday off, and separate weekday off, are enumerated from one to five, respectively (Table 1).

Each schema (out of 63 schemata) has a predetermined cost, obtained by the multiplication of its off day type enumeration and shift type enumeration. An illustrative cost calculation example for some of the schemata is given in Table 2. One can easily recognize that more preferable schemata have lower cost when the schema cost calculation is examined. So, the objective of the problem turns out to be minimizing the cost. As an example, "DDDDDXX" is the most preferred and lowest cost tour.

A small sample requirement input for four employees is illustrated in Table 3. Schema ID is schema identifier; a schema may be identified by a number between one and 63. FOSSP scheduling decides schema usage quantities by caring for the employee requirements in each time period. The requirements for Monday day shift and evening shift are zero and no employees are assigned to day shift on Monday. The requirement for Monday night shift is one and one employee will be assigned to night shift on Monday with schema ID 63. FOSSP output satisfies employee requirements by selecting the most suitable schema with the required usage quantity.

Since the requirements are the same for all weeks during the planning period (four weeks in our case), the same usage quantity of each schema must be used for each week. The sample output of FOSSP is given in Table 4; schema ID 1 is assigned to employee one and employee

Table 1: Shift type and off day type enumeration.

Type	Cost
Day shift	1
Evening shift	2
Night shift	3
Weekend off	1
Friday-Saturday off	2
Sunday-weekday off	3
Consecutive weekday off	4
Separate weekday off	5

Table 2: Some examples regarding schema cost computation.

Schema	String	c (s)
21	DDDDDXX	1 (1 ∗ 1)
42	EEEEEXX	2 (1 ∗ 2)
63	NNNNNXX	3 (1 ∗ 3)
8	DXDXDDD	5 (5 ∗ 1)
29	EXEXEEE	10 (5 ∗ 2)
50	NXNXNNN	15 (5 ∗ 3)

Table 3: Sample requirement input.

Time slot ID	Definition	Requirement
1	Monday day shift	0
2	Monday evening shift	0
3	Monday night shift	1
4	Tuesday day shift	1
5	Tuesday evening shift	1
6	Tuesday night shift	1
7	Wednesday day shift	1
8	Wednesday evening shift	1
9	Wednesday night shift	1
10	Thursday day shift	2
11	Thursday evening shift	1
12	Thursday night shift	1
13	Friday day shift	2
14	Friday evening shift	0
15	Friday night shift	1
16	Saturday day shift	2
17	Saturday evening shift	1
18	Saturday night shift	0
19	Sunday day shift	2
20	Sunday evening shift	1
21	Sunday night shift	0

three in week one, two other employees in week two, and so on. Schema ID 2 is not assigned to anyone. Schema ID 25 is assigned to one employee and schema ID 63 is assigned to one employee in each week. The problem assigns the same schemas with same quantities to different employees for given number of weeks.

3.3. Fairness Oriented Shift Scheduling Problem Model. The scheduling problem we handled in this study was modeled as two consecutive problems in previous studies. Tour scheduling problem and employee assignment problem [20] are jointly sufficient to define this particular scheduling issue. Note that tour scheduling problem reduces to set covering problem [13] which is NP-Hard. Similarly employee assignment problem reduces to the generalized assignment problem [14] which is also shown to be NP-Hard. Since FOSSP corresponds to the combination of both problems, it is also NP-Hard. Solving the problem as two subproblems offers suboptimal solutions; however, whole parts of the issue are significant and we need to find a solution for an integrated single problem. FOSSP is the first sample that put those problems together and solves them as a unique problem in literature, to the best of our knowledge. In the following, we provide the necessary notation and then the mathematical formulation of FOSSP.

Table 4: Schema and employee assignments for four weeks.

Employee ID	Week 1	Week 2	Week 3	Week 4
1	1	1	25	63
2	63	25	1	1
3	1	63	1	25
4	25	1	63	1

x_{ws} is the number of times schema s is used for each week w,

S is the set of schemas (remember that a schema is a seven-lettered string denoting the shift types and off days; each schema is denoted by a number from 1 to 63),

T represents the time slots of week (there are totally 21 time slots in a week; each day contains three shifts (planning time slot), and seven days comprise 21 time slots similarly; each time slot is denoted by a number from 1 to 21),

E is the set of employees,

W is the set of weeks,

c_s is the cost of schema s,

r_t is the requirement (number of required workers) at time slot t,

$$A_{ews} = \begin{cases} 1, & \text{if employee } e \text{ assigned to schema } s \text{ in week } w, \\ 0, & \text{otherwise,} \end{cases} \tag{1}$$

A_{ews} is the assignment variable and shows the assignments of each employee e for each week w and for each schema s. A is equal to one if an employee e is assigned to schema s in week w, and it is equal to zero otherwise,

$$b_{st} = \begin{cases} 1, & \text{if schema } s \text{ covers times lot } t, \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

b_{st} indicates the time slot coverage of each schema. If schema s comprises time slot t then b is equal to one and it is equal to zero otherwise. $M = \sum_s \sum_w \sum_e A_{ews} * c_s/|E|$ is the average cost value for all employees, calculated by total of schema assignment number and schema cost multiplication, divided by the number of employees,

$N_e = \sum_w \sum_s A_{ews} * c * c_s$ is the all weeks' total cost value. This value is calculated separately for each employee. If employee e is assigned to schema s during week w, A_{ews} variable is equal to one and schema cost is added to the total. The sum of all weeks' schema costs is calculated and divided by total number of planning week number W. So average cost value is calculated for each employee,

$D_e = (M - N_e)^2$ is the sum of squared distances of each employee individual total N_e and the general mean total M. The sum of square deviations which measures the pairwise differences is one of the inequality indices widely applied in the literature [41, 42],

$$\text{minimize} \sum_e D_e, \tag{3}$$

$$\sum_s A_{ews} = 1, \quad \text{for } \forall e, w, \tag{4}$$

$$\sum_e A_{ews} = x_{ws}, \quad \text{for } \forall w, s, \tag{5}$$

$$\sum_s x_{ws} b_{st} \geq r_t, \quad \text{for } \forall w, t. \tag{6}$$

The objective is minimizing the squared distances of each employee so that maximum fairness is sought. Constraint (4) means that each employee is assigned to only one schema in each week. Constraint (5) ensures that the total number of employees assigned to schema s for each week is equal to x_{ws} (number of total employee assignments in each week and for each schema). Constraint (6) ensures that requirement in each time slot t (i.e., the required number of employees) is satisfied.

$$A_{ews} * b_{s(21)} + A_{e(w+1)(s_1)} * b_{(s_1)1} \leq 1, \quad \text{for } \forall e, w, s, s1, \tag{7}$$

$$\sum_{k=16}^{21} A_{ews} * b_{sk} + \sum_{k=1}^{3} A_{e(w+1)(s1)} * b_{(s_1)k} \geq 1, \quad \text{for } \forall e, w, s, s1, \tag{8}$$

$$\sum_{k=3q+1}^{21} A_{ews} * b_{sk} + \sum_{k=1}^{3q} A_{e(w+1)(s_1)} * b_{(s_1)k} \leq 6, \quad \text{for } q = 0 \ldots 6 \text{ for } \forall e, w, s, s1, \tag{9}$$

$$s, s1 \in S, \quad t, k \in T, \quad e \in E, \quad w, w+1 \in W. \tag{10}$$

Variables $w + 1$ and $s1$ are used in constraints five to seven. $w + 1$ demonstrates the following week and $s1$ corresponds to the schema assignment of the following week.

Additional constraints (equations (7)–(9)) may be explained as follows: Constraint five carries out the rule; if employee has night shift on Sunday, (s)he cannot be assigned to day shift on adjacent Monday. Time slot 21 corresponds to Sunday night shift and time slot 1 corresponds to Monday day shift (Table 3). This constraint prevents assigning any two employees to adjacent shifts for consecutive weeks w and $w + 1$. Another legal regulation is that if an employee has weekend off, he/she cannot be off on Monday (the first day of the next week), implemented by constraint six. Time slot numbers from 16 to 21 embrace weekend period, while time slots 1 to 3 correspond to three shifts of Monday. Total assignment for those three days must be greater than or equal to one. Hence, if someone had weekend off, that person will work on Monday of the next week ($w + 1$), or in other words an employee cannot be off for three consecutive days. Constraint seven enforces the rule; an employee cannot work for more than six days consecutively. If q is equal to zero, it checks only the first week; if q is equal to one, it calculates the total for six days of first week starting from Tuesday and one day (Monday) from second week and so on.

The proposed model is implemented using GAMS software. Tests are done in GAMS for small dataset and the results are demonstrated in Section 6. However, for the large-scale real-world requirements, we have implemented heuristic based solution techniques and developed a novel hybrid technique described in detail in the next section.

4. Proposed Solution Methods

In this section, the solution methods are clarified. Initial individual generation process for FOSSP, hyperheuristics, simulated annealing, migrating birds optimization algorithms, and a novel hybrid algorithm, which is one of the most important contributions of this study, are explained, respectively.

4.1. Initial Individual Generation for FOSSP. Recall from the small example that the ultimate output of FOSSP is a matrix containing schema numbers assigned to employees for each week. Therefore, a solution (also called an individual) to a problem instance will refer to a feasible output in the remainder of the manuscript. Again, recall that, due to the nature of the problem, same set of schema numbers are used in each week with different permutations. A sample FOSSP individual is illustrated in Table 5 for ten employees and four weeks. Employee number one is assigned to schema number five in week one, employee number two is assigned to schema twenty in week one, and so on.

We want to remind that requirement list is updated periodically before preparing the schedules and the requirements are the same for all weeks in planning horizon. Initial individual creation progress of FOSSP is as follows: Schemas are assigned to employees for the first week satisfying the requirements in each time interval. The schedules of subsequent weeks (week two, week three, etc.) are constructed by shuffling assigned schema values in week one.

Table 5: Sample four-week, ten-employee FOSSP individual.

Week	e1	e2	e3	e4	e5	e6	e7	e8	e9	e10
1	5	20	30	35	40	53	58	8	18	27
2	8	18	27	30	35	40	53	58	5	20
3	8	18	27	53	58	40	30	35	5	20
4	40	53	27	30	35	8	18	58	5	20

Additional controls for meeting the requirements and controlling official rules are also taken into account.

4.2. Hyperheuristics. Hyperheuristics (HH) is a neighborhood search technique used for selecting, applying, and managing low-level heuristics and it is widely used in solving combinatorial optimization problems [43]. Pseudocode of the HH is presented in Algorithm 1. HH has two important functionalities, one of which is heuristic selection mechanism and the other is the move acceptance mechanism. Heuristic selection mechanism is responsible for selecting a low-level heuristic within the pool. Move acceptance mechanism decides whether to accept new solutions or not. In our implementation, simple random heuristic selection, random permutation heuristic selection, and adaptive searching heuristic selection strategies are used. Only Improvement and Monte Carlo acceptance strategies are exploited as acceptance strategies. Now let us explain the low-level heuristics applied for tackled problems, respectively.

4.2.1. Low-Level Heuristics for FOSSP. There are totally four low-level heuristics for FOSSP in this study. Each heuristic checks the additional constraints specified in the model.

Heuristic 1 (H1): H1 selects a random schema among the ones that are already assigned to an employee. Another random schema is selected with tournament selection with size five. Five schemas are selected randomly among 63 schemas, and one of those five schemas is chosen as second schema. Tournament selection increases the possibility of choosing a lower cost schema.

Heuristic 2 (H2): H2 randomly selects two employees for each week and swaps their schema assignment values.

Heuristic 3 (H3): H3 selects two employees randomly for each week and inserts the schema value in subsequent location to the next place in prior employee location.

Heuristic 4 (H4): H4 selects two employees randomly for each week and inverts schema assignment values between these employee locations.

4.2.2. Low-Level Heuristics for QAP. Four low-level heuristics are implemented on QAP. Those may be explained basically as follows:

Swap: Swap heuristic simply selects two points in a QAP individual and relocates their places.

Insert: Two random points are selected within an individual and the subsequent one is placed right after the first random point. All points between selected points move one step to the right.

Inverse: Two random points are selected within an individual. All assignments between the selected points are reversed.

Scramble: Two random points are selected on an individual. Assignments between the selected points are scrambled.

4.2.3. Heuristic Selection Strategies. Selecting the low-level heuristics requires a strategy. We applied three different heuristic selection strategies. These are simple random (SR), random permutation (RP), and adaptive searching (AS).

(i) SR randomly selects one of the low-level heuristics and applies it only once.

(ii) RP firstly constructs a permutation of low-level heuristics and applies each one once according to the permutation.

(iii) AS is a smart selection mechanism that we designed in this study. Each low-level heuristic has an equal initial score (Ri) and those scores are updated during the run. If heuristic improves the solution, its score is increased as much as the change quantity (Cq); if it worsens the solution, its score is decreased as much as Cq. A successful heuristic's score may not exceed predefined upper boundary (Ub); analogously an abortive heuristic's score may not be less than the predefined lower boundary (Lb). The heuristic with greater score has more probability to be selected. It is applied once and the scores are updated according to its performance.

4.2.4. Acceptance Strategies. Once a low-level heuristic is selected and applied, the acceptance of new solution also requires a strategy. We applied two strategies; these are Only Improvement (OI) and Monte Carlo (MC).

(i) OI acceptance strategy only accepts better solutions.

(ii) MC acceptance strategy accepts better solutions certainly. Additionally, it accepts worse solutions with a small probability (Monte Carlo constant) in order not to get stuck at the local optima.

4.3. Simulated Annealing. Simulated annealing (SA) is a well-known local search algorithm proposed by Kirkpatrick et al. [44], inspired from the annealing process of metal work. A starting temperature T is determined. The larger this value is, the more inferior solutions are encouraged. After a predetermined number of iterations, T is set to T/a. When a is large, the temperature decrease ratio is large and the acceptance of inferior solutions become less likely at a greater rate. When temperature becomes less than a predefined value, SA stops (the stopping condition). The number of iterations at each temperature is limited by R and

```
Initialization i = 0
Initial candidate solution S_0
Final solution S_f
sNew = S_0, S_f = S_0
while (i < maxIterations) Select heuristic according to heuristic selection strategy
   sNew = applySelectedHeuristic (S_f)
   if (sNew is accepted according to move acceptance strategy)
      S_f = S_new
   end if
   i++
end while
return S_f
```

Algorithm 1: Pseudocode of HH.

greater values of R correspond to slower cooling; that is, more neighbor solution trial is occurring when there is a greater likelihood of inferior exchanges being accepted. In our implementation, R value changes and increases by an initially specified b value. When a random neighbor is generated, it is compared with the previous solution. If it is better than the previous one, it is accepted. If it is worse than the previous one, it may still be accepted with a small probability. This probability is calculated based on the difference of old and new solution quality and the temperature. The pseudocode of SA is given in Algorithm 2.

4.4. Migrating Birds Optimization. Migrating birds optimization (MBO) is a neighborhood search technique proposed by Duman et al. [40], inspired from the V formation flight of the migrating birds. It has been recently proposed but successfully applied in various research areas such as task allocation problem [45], flow shop scheduling problem [46–48], U-shaped assembly line balancing problems with workers assignment [49], and continuous functions [50].

The pseudocode of MBO is given in Algorithm 3. Similar to birds of a V shape migration, some initial solutions organize a V formation including one leading solution and some followers. In the flock of solutions, a limited number of neighboring solutions for each main solution are generated. The neighboring solutions of each initial solution are evaluated and if there are any improvements among them, that initial solution is replaced by the solution provided by the most improved neighbor. Then, each main solution is tried to be improved further by the help of its neighbors. This means that each solution will share some of its unused neighbors to the next (behind) main solution. Therefore, except the leading solution, the other main solutions of the flock have the chance to be improved by one of the neighbors of the main solution in front of them. The procedure is repeated a number of times. Then, the leading solution moves to the end of the flock and one of its followers becomes the new leader. The same procedure is done and repeated for the new flock. The algorithm continues until a number of iterations are reached. Finally, the best solution of the flock is introduced as the solution of the MBO algorithm.

4.5. A New Hybrid Algorithm: HHMBO. Now, we present a novel hybrid technique by integrating various hyperheuristic techniques with the migrating birds optimization algorithm. To the best of our knowledge, this work is the first attempt of hybridizing MBO with hyperheuristics.

MBO is a metaheuristic algorithm that explores new solutions by single step heuristics as illustrated in Figure 1. Neighbor generation technique is fixed and does not change throughout the execution of the algorithm. Therefore the performance of classical MBO is much dependent on the neighbor generation technique. However, giving chance more than one heuristic with different characteristics to discover search space may increase its exploration capability. Based on this idea, we integrated the recently presented MBO algorithm with the hyperheuristic principles. Instead of a single discovery method, we took the advantage of exploiting a set of low-level heuristics in order to construct the neighbor solutions in the population.

Neighborhood generation process of hyperheuristics embedded migrating bird optimization (HHMBO) is presented in Figure 2. The neighbor solution generation part of MBO is hybridized by HH characteristics, which is capable of managing heuristic pool. A set of heuristic selection strategies are handled for choosing the low-level heuristics. In our implementation, we have applied simple random, random permutation, and adaptive searching heuristic selection strategies. The new neighboring solution is accepted or rejected according to the current move acceptance specifications. We have implemented Only Improvement and Monte Carlo move acceptance strategies. HHMBO operates in different ways according to the adjusted heuristic selection strategy and the move acceptance strategy. As a result, the HHMBO types that emerged in the application can be listed as follows: simple random Only Improvement (SR_OI), simple random Monte Carlo (SR_MC), random permutation Only Improvement (RP_OI), random permutation Monte Carlo (RP_MC), adaptive searching Only Improvement (AS_OI), and adaptive searching Monte Carlo (AS_MC). HHMBO generates neighbor solutions by one of those hyperheuristic variations. The pseudocode of the algorithm is given in Algorithm 4.

HHMBO offers a collaboration of hyperheuristics and migrating birds optimization algorithm. HH provide an

```
Generate a random initial solution and indicate it as current solution, cs, initialize
temperature T
Best solution(bs) = cs
while termination condition is not satisfied
  for R times
    Obtain a neighbor solution, ns of cs
    delta = cost of ns−cost of cs
    if delta < 0
      cs = ns
    else if random() < e^(|delta|/T)
      cs = ns
    if cost of ns < cost of bs
      bs = ns
  end for
    T = T/a
    R = R * b
end while
```

Algorithm 2: Pseudocode of SA.

```
Generate nob initial solutions randomly and place them on a hypothetical V formation arbitrarily
while termination condition is not satisfied
  for nof times
    Try to improve leading solution by generating and evaluating non neighbors of it
    for each solution s_i in the flock (except leader)
    Try to improve s_i by evaluating (non-olf) neighbors of it and olf unused best neighbors from the solution in the front
    end for
  end for
  Move the leader solution to the end and forward one of the solutions following it to the leader position
end while
```

Algorithm 3: Pseudocode of MBO.

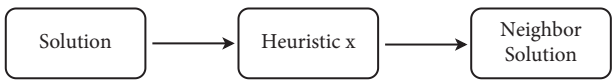


Figure 1: Neighborhood generation in classical MBO.

extensive exploration capability due to their ability to quickly adapt according to the problem instance at hand. MBO exploits the leader solution more detailed each time and the leader is updated in each generation. HHMBO takes the advantage of exploring the solution space by the exploration talent of HH in addition to the exploitation skills of MBO. The properties of the HHMBO which distinguish it from the other metaheuristic approaches are listed as follows:

(i) A number of solutions running in parallel

(ii) The benefit mechanism among the solutions

(iii) Exploiting the leader solution more than the others in each generation

(iv) Exploring the search space by multiple operators where the best strategy is determined by HH

5. Experimental Setup

In this section, we provide the details of computational experiment settings. Firstly the dataset used in experiments is described. Then the parameter settings of MBO, HH, SA, and HHMBO are explained. Lastly, the benchmark dataset is clarified.

GAMS is used to verify and test the mathematical model. All algorithm implementations are done using Java programming language on Eclipse IDE running on Intel i7-5500U CPU and 16 GB memory computer operated by Windows 10 operating system. Besides the basic constraints on the formulation presented, additional legal regulation constraints have also been applied while implementing each solution technique.

5.1. Data Preparation. We had a limited amount of real data provided by our collaborator company. However, we needed a great amount of experimental data to achieve a fair comparison among solution techniques we had implemented. Therefore, a simple application is prepared, which generates synthetic data, which have the general characteristics of the real data. Requirements are selected randomly

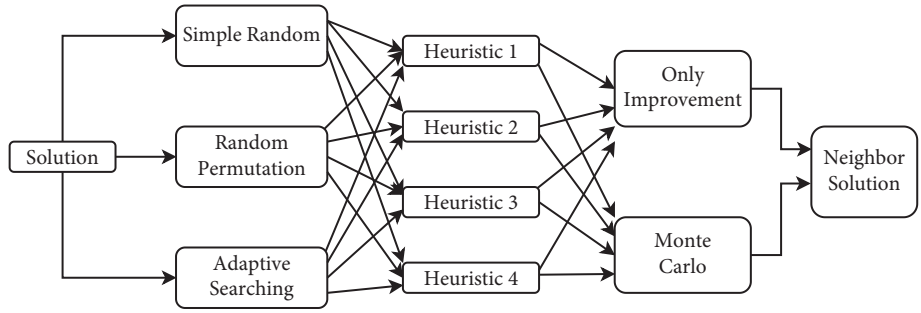


Figure 2: Neighborhood generation mechanism in HHMBO.

Generate *nob* initial solutions randomly and place them V formation
Determine the heuristic selection strategy (Simple random, random permutation or adaptive searching)
Determine the move acceptance strategy (Only improvement or monte carlo)
while termination condition is not satisfied
  for *nof* times
    Try to improve leading solution by generating and evaluating *non* neighbors of it applying selected low level heuristic
    for each solution s_i in the flock (except leader)
      Try to improve s_i by evaluating (*non-olf*) neighbors of it and *olf* unused best neighbors from the solution in the front
      Accept or reject the neighbor solution according to current move acceptance strategy.
    end for
  end for
  Move the leader solution to the end and forward one of the solutions following it to the leader position
end while

Algorithm 4: Pseudocode of HHMBO.

among a certain range, where lower and upper boundaries are calculated based on employee number. A real input dataset for 200 employees and sample synthetic requirement data values are illustrated in Table 6. The employee numbers are shown in the top row and they vary from 30 to 500. This illustration comprises a sample requirement input for varied number of employees. First column of the table contains the time slot IDs; each one corresponds to a day and a shift type. The requirements in each time slot for a company with 200 (real), 30, 50, 100, 150, 200, and 500 employees are given in the subsequent columns.

5.2. Parameter Settings. MBO and SA perform a unique heuristic technique for improving solutions. Since we have four different heuristic techniques, we have conducted an experiment to understand which of the four operators in the system is compatible with which algorithm. In order to assess a fair comparison among all applied techniques, we set the stopping criterion of the algorithms based on the number of instances (noi) created while each algorithm is running. This experiment is done on each of the datasets, while the number of instances (noi) is set to 10,000 and each case is repeated 50 times. Heuristic operators are listed from the best performing one to the least performing one in Table 7. H1 through H4 correspond to heuristic one through heuristic four, respectively, which are explained in Section 4.2.1. Swap, Insert, Inverse, and Scramble are applied on QAP and explained in Section 4.2.2.

In order to get the best performance from the algorithms for FOSSP, we need to use their best performing values. These best performing values are discovered with fine-tuning experiments. The number of solutions that algorithm can generate while surfing in the solution space is limited to 50,000 for each algorithm. The experiments are repeated ten times and the results acquired from the average performances are represented in Tables 8 and 9. Best parameter values are emphasized in bold font.

5.3. Experimental Setup of Benchmark Data. Apart from the specific problem defined in this study, computational experiments are also conducted with QAP to measure the success of the HHMBO. Computational experiments presented have been done using QAPLIB benchmark dataset, where QAP instances are available together with their best known solutions [51]. Fine-tuning experiments are handled to find out the best performing values. The number of solutions that the algorithm can generate while navigating the solution space is limited to 50,000. The experiments were repeated ten times and the results from the average performances are shown in Table 10, with the best parameter values being in bold.

6. Results and Discussion

In this section, we present and analyze the results of computational experiments. GAMS results are presented firstly. Then, the results obtained by heuristic solution

Table 6: Sample real and synthetic inputs.

ID	Time slot meaning	Real data 200	Synthetic 30	Data 50	100	150	200	500
1	Monday day shift	58	3	12	15	33	12	36
2	Monday evening shift	36	8	11	2	16	12	30
3	Monday night shift	50	8	7	23	9	42	76
4	Tuesday day shift	58	5	6	20	28	10	2
5	Tuesday evening shift	36	9	10	4	17	7	60
6	Tuesday night shift	50	9	7	1	1	34	41
7	Wednesday day shift	58	6	9	13	25	29	94
8	Wednesday evening shift	36	5	1	16	26	4	40
9	Wednesday night shift	5	3	4	14	15	40	3
10	Thursday day shift	58	6	8	6	28	40	88
11	Thursday evening shift	36	3	11	16	25	27	31
12	Thursday night shift	50	7	5	19	30	12	78
13	Friday day shift	58	8	7	5	33	2	103
14	Friday evening shift	36	1	4	14	16	17	89
15	Friday night shift	50	8	6	17	6	26	48
16	Saturday day shift	2	6	10	2	14	24	80
17	Saturday evening shift	5	3	11	1	24	1	96
18	Saturday night shift	5	2	2	5	15	10	79
19	Sunday day shift	5	1	13	1	1	39	55
20	Sunday evening shift	49	1	11	23	26	35	56
21	Sunday night shift	36	5	10	22	14	43	40

Table 7: Sample real and synthetic inputs.

FOSSP	Low-level heuristics	QAP	Low-level heuristics
MBO	**H2**, H4, H3, H1	MBO	**Swap**, Insert, Inverse, Scramble
SA	**H1**, H3, H4, H1	SA	**Swap**, Insert, Inverse, Scramble

Table 8: Parameter fine-tuning experiment results for FOSSP.

Parameter	MBO	HHMBO					
		SR_OI	SR_MC	RP_OI	RP_MC	AS_OI	AS_MC
nob	5, 21, **51**	5, 21, **51**	5, 21, **51**	5, 21, **51**	5, 21, **51**	5, 21, **51**	5, 21, **51**
non	**3**, 5, 7	3, 5, **7**	3, **5**, 7	3, **5**, 7	3, **5**, 7	3, **5**, 7	**3**, 5, 7
nof	5, **10**	**5**, 10	**5**, 10	5, **10**	**5**, 10	5, **10**	**5**, 10
olf	**1**, 2, 3	**1**, 2, 3	**1**, 2, 3	1, **2**, 3	**1**, 2, 3	**1**, 2, 3	1, **2**, 3
Monte Carlo constant			0.0005, **0.001**, 0.005		0.0005, 0.001, **0.005**		0.0005, 0.001, **0.005**

Table 9: Parameter fine-tuning experiment results for FOSSP.

Parameter	SA	Parameter	Adaptive Searching
T	**50.000**, 75.000, 100.000	*Ri*	10, **15**, 20
A	**1.1**, 1.2, 1.3	*Cq*	0.5, 0.8, **1**
R	10, **20**, 30	*Ub*	**40**,50,60,100
b	**1.1**, 1.2, 1.3	*Lb*	**0**, 5, 10

techniques on FOSSP are presented. Lastly, experimental outcomes on QAPLIB are presented.

6.1. GAMS Results. The GAMS utilizing CPLEX solver is used to verify and test the mathematical model of FOSSP. If the schedule is completely fair and all assignments are scrupulous for each employee, then fitness is equal to zero. The result gets away from zero, while the solution wanders away from fairness.

Solver tests are done for datasets with 10, 20, 30, and 60 employees and results are illustrated in Table 11 in terms of cost and execution time. FOSSP is capable of finding optimal value (zero) for employee numbers 10 to 30. FOSSP produces results closer to optimal for employee number 60. As the size of the problem gets larger, complexity increases. Obviously, execution time increases beyond linear behavior as the number of employees increases. Model attains border line of maximum iteration number in GAMS. Execution is terminated on maximum iteration number; hence, the

Table 10: Parameter fine-tuning experiment results for QAP.

Parameter	MBO	HHMBO						Parameter	Adaptive searching
		SR_OI	SR_MC	RP_OI	RP_MC	AS_OI	AS_MC		
nob	5, 21, **51**	5, 21, **51**	5, 21, **51**	5, 21, **51**	5, 21, **51**	5, 21, **51**	5, 21, **51**	*Ri*	10, 15, **20**
non	**3**, 5, 7	**3**, 5, 7	3, **5**, 7	**3**, 5, 7	3, **5**, 7	**3**, 5, 7	**3**, 5, 7	*Cq*	0.5, **0.8**, 1
nof	5, **10**	5, **10**	**5**, 10	**5**, 10	5, **10**	5, **10**	**5**, 10	*Ub*	40,50,**60**,100
olf	**1**, 2, 3	**1**, 2, 3	**1**, 2, 3	**1**, 2, 3	**1**, 2, 3	**1**, 2, 3	**1**, 2, 3	*Lb*	**0**, 5, 10
Monte Carlo constant			0.0005, **0.001**, 0.005		**0.0005**, 0.001, 0.005		**0.0005**, 0.001, 0.005		

Table 11: GAMS results for FOSSP datasets.

Number of employees	Cost values	Execution time (minutes)
10	0	12.824
20	0	500.334
30	0	968.884
60	3.34	988.58

Table 12: Average results when noi is N^2.

Employees number	MBO	SA	HHMBO						HH					
			SR_OI	SR_MC	RP_OI	RP_MC	AS_OI	AS_MC	SR_OI	SR_MC	RP_OI	RP_MC	AS_OI	AS_MC
30	21.81	33.67	16.95	**13.13**	13.79	20.16	19.77	19.67	20.91	21.24	19.88	22.48	20.17	19.99
50	29.79	59.82	24.59	**15.72**	22.80	21.72	30.48	36.84	27.78	29.94	30.83	30.33	42.69	38.23
100	52.13	161.54	44.99	44.33	**39.99**	51.48	63.09	68.60	46.07	48.96	54.99	110.15	135.29	108.24
150	62.11	292.00	61.51	67.18	**57.00**	63.09	98.14	101.93	91.42	178.29	133.70	135.24	208.15	244.27
200	70.12	367.25	67.99	73.20	**67.67**	71.78	143.12	141.09	241.61	260.86	315.59	209.68	248.82	350.38
500	287.60	926.11	301.62	254.06	**247.60**	257.94	499.58	460.58	962.85	939.88	588.76	765.77	712.36	912.62

Table 13: Standard deviations of the experiment.

Employees number	MBO	SA	HHMBO						HH					
			SR_OI	SR_MC	RP_OI	RP_MC	AS_OI	AS_MC	SR_OI	SR_MC	RP_OI	RP_MC	AS_OI	AS_MC
30	2.58	2.55	2.05	2.32	2.07	2.13	2.03	**2.01**	3.67	3.51	3.65	3.96	3.96	2.89
50	1.67	1.87	1.59	1.65	**1.52**	1.64	**1.52**	1.58	2.15	3.02	2.33	2.83	2.78	3.36
100	1.78	2.00	1.95	1.80	**1.71**	1.88	1.75	1.77	3.51	3.16	3.68	3.44	3.41	3.28
150	1.73	1.92	1.92	1.69	**1.52**	1.77	1.90	1.89	2.99	3.33	3.31	3.18	3.46	3.20
200	1.96	1.75	2.09	2.12	**1.48**	2.01	2.09	2.01	2.96	2.85	2.71	3.05	2.63	2.75
500	1.27	1.28	1.43	1.52	**1.22**	1.32	1.37	1.34	1.99	2.11	2.05	1.91	2.15	2.14

execution time values are the total minutes required for maximum iteration in each case.

GAMS is capable of solving the model, but it is poor in terms of execution time, especially for large-scale problems. Therefore, we needed to implement and use heuristic based solution techniques for obtaining reasonable results within acceptable run time.

6.2. Results Obtained by Utilizing Synthetic FOSSP Dataset. In this subsection, results of heuristic based solution techniques on FOSSP are presented. If N is the number of employees, we limited the run time with N^2 iterations in order to assess a fair comparison among the applied techniques. This corresponded to 0.9 *s* for a problem of size 30 and to 34 *s* for a problem of size 100 on the specified machine. Consequently, as the complexity of the problem increases, the effort to be spent on its resolution also increases.

Solution techniques are tested for numbers of employees of 30, 50, 100, 150, 200, and 500. We have generated 100 sample synthetic inputs for each case. Average numerical cost values gained from the ten runs are demonstrated in Table 12. For all numbers of employees, HHMBO variants discover the best results. For numbers of employees of 30 and 50, HHMBO SR_MC variant outperforms the others by at least 5% on average. When the number of employees increases to 100, 150, 200, and 500, HHMBO RP_OI variant outperforms the other techniques.

When the performances are analyzed in accordance with the increase of the number of employees, it is apparent that MBO and HHMBO variants retain their accomplishment as complexity rises. However, MBO is 66%, 90%, 30%, 9%, and 4% worse on average than the best performing HHMBO variant for numbers of employees of 30, 50, 100, 150, and 200, respectively. It is clear that both algorithms are successful in spite of increasing problem complexity. However,

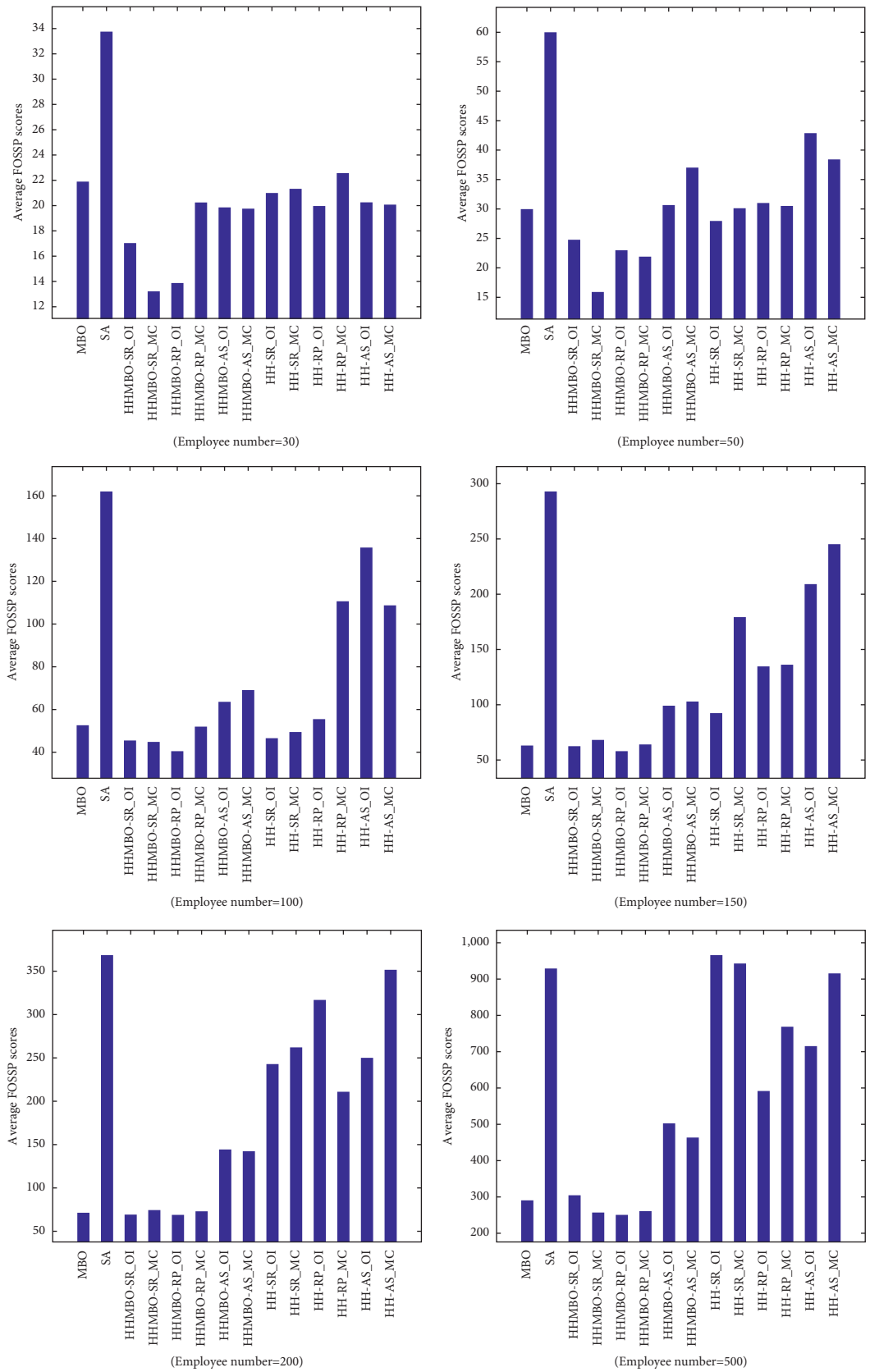


Figure 3: The performance of each algorithm for each number of employees.

Table 14: The t-test results of HHMBO-RP_OI and the state-of-the-art algorithms on FOSSP.

Algorithm 1 ↔ Algorithm 2	30	50	100	150	200	500
HHMBO-RP_OI ↔ MBO	$1.3E-02$	$9.9E-02$	$3.1E-04$	$1.7E-02$	$5.2E-05$	$3.7E-03$
HHMBO-RP_OI ↔ SA	$1.6E-05$	$3.8E-04$	$5.7E-05$	$6.9E-06$	$4.1E-08$	$4.9E-04$
HHMBO-RP_OI ↔ HH-SR_OI	$2.4E-06$	$7.4E-05$	$1.1E-08$	$3.4E-12$	$2.2E-11$	$2.7E-22$
HHMBO-RP_OI ↔ HH-SR_MC	$9.7E-06$	$1.5E-04$	$2.4E-10$	$3.3E-11$	$4.0E-15$	$2.2E-19$
HHMBO-RP_OI ↔ HH-RP_OI	$2.6E-05$	$1.3E-02$	$1.9E-11$	$1.4E-11$	$5.8E-14$	$2.7E-23$
HHMBO-RP_OI ↔ HH-RP_MC	$2.8E-04$	$1.9E-02$	$6.4E-10$	$7.7E-09$	$3.9E-12$	$1.4E-21$
HHMBO-RP_OI ↔ HH-AS_OI	$4.1E-03$	$3.8E-02$	$1.7E-05$	$4.9E-12$	$7.8E-16$	$1.4E-17$
HHMBO-RP_OI ↔ HH-AS_MC	$6.7E-04$	$1.2E-03$	$2.9E-06$	$6.0E-12$	$2.2E-15$	$6.1E-18$

Table 15: The percentage deviations of heuristic results from the best solution.

Problem	Density	Size	BKS	MBO	HHMBO					
					SA_OI	SR_MC	RP_OI	RP_MC	AS_OI	AS_MC
esc64a	3.17	64	116	0.0	0.0	0.0	58.6	58.6	0.0	0.0
tai64c	4.13	64	1,855,928	0.0	0.0	0.0	11.4	10.1	0.0	0.1
chr22a	8.68	22	6,156	9.6	16.8	13.3	41.2	45.8	8.1	7.1
esc16j	9.38	16	8	0.0	0.0	0.0	0.0	25.0	0.0	0.0
chr20a	9.50	20	2,192	31.3	55.3	56.6	147.7	132.6	27.9	20.1
chr20b	9.50	20	2,298	25.7	35.9	47.5	67.6	119.4	17.8	36.3
chr18a	10.49	18	11,098	54.2	71.0	81.6	197.2	190.1	53.8	55.2
esc16i	11.72	16	14	0.0	0.0	0.0	28.6	14.3	0.0	0.0
chr15a	12.44	15	9,896	22.6	12.7	40.4	76.2	116.9	35.8	30.1
chr15b	12.44	15	7,990	38.2	70.5	37.7	193.3	168.5	33.4	51.4
chr15c	12.44	15	9,504	53.8	39.2	62.9	95.4	96.6	39.2	42.0
chr12c	15.28	12	11,156	14.9	28.1	14.3	60.4	46.0	18.8	8.5
chr12a	15.28	12	9,552	20.5	37.5	19.7	60.2	58.5	19.8	5.7
esc16d	16.41	16	16	0.0	0.0	0.0	12.5	0.0	0.0	0.0
esc16e	16.41	16	28	0.0	0.0	0.0	14.3	14.3	0.0	0.0
esc16g	16.41	16	26	0.0	0.0	0.0	15.4	23.1	0.0	0.0
esc32d	17.58	32	200	0.0	3.0	4.0	37.0	35.0	0.0	0.0
esc32c	25.59	32	642	0.0	0.0	0.0	13.7	14.0	0.0	0.0
esc32h	27.54	32	438	0.0	2.3	2.3	26.5	21.0	0.0	0.0
esc16a	29.69	16	68	0.0	0.0	0.0	8.8	8.8	0.0	0.0
scr15	37.33	15	51,140	7.0	11.3	7.0	29.4	28.1	6.9	9.2
esc16c	39.84	16	160	0.0	0.0	0.0	5.0	8.8	0.0	0.0
esc16b	71.88	16	292	0.0	0.0	0.0	0.0	0.0	0.0	0.0
lipa20b	89.25	20	27,076	11.2	14.8	17.2	22.0	21.9	9.8	11.0
esc16h	89.84	16	996	0.0	0.0	0.0	1.6	0.0	0.0	0.0
lipa20a	90.25	20	3,683	2.2	3.0	3.0	4.5	4.5	2.4	2.7
had12	91.67	12	1,652	0.4	0.7	0.1	3.4	2.4	1.0	0.8
rou12	91.67	12	235,528	3.0	4.7	2.8	9.8	9.0	5.7	4.9
rou15	93.33	15	354,210	5.1	7.7	6.8	13.7	13.4	5.0	4.8
nug16b	93.75	16	1,240	3.9	6.9	3.4	12.9	13.7	4.5	4.4
els19	94.74	19	17,212,548	4.7	8.7	5.5	39.7	47.3	1.8	2.7
lipa40b	94.81	40	476,581	5.3	21.0	19.6	26.0	26.3	3.5	6.5
lipa40a	95.06	40	31,538	1.3	2.0	2.0	2.9	2.8	1.6	1.6
lipa50b	95.88	50	1,210,244	17.9	21.8	21.7	25.3	25.5	8.3	11.2
tai60b	98.33	60	608,215,054	1.1	2.2	2.7	40.2	41.0	0.6	1.1
sko100a	99.00	100	152,002	0.7	5.9	5.7	13.9	13.9	0.7	2.6
bur26b	100.00	26	3,817,852	0.1	0.5	0.3	2.5	3.0	0.0	0.1
bur26a	100.00	26	5,426,670	0.1	0.4	0.4	2.8	1.8	0.1	0.2
bur26c	100.00	26	5,426,795	0.0	0.2	0.2	3.1	3.1	0.0	0.1
bur26e	100.00	26	5,386,879	0.0	0.3	0.4	3.4	3.6	0.0	0.1
bur26g	100.00	26	10,117,172	0.0	0.1	0.4	2.6	3.0	0.1	0.0
Average				8.2	11.8	11.7	34.9	35.9	**7.5**	7.8

HHMBO variants obtain slightly better results than MBO in each condition.

The best performing HHMBO variant is 156%, 281%, 304%, 412%, and 443% better than SA for numbers of employees of 30, 50, 100, 150, and 200, respectively. As the complexity of the problem increases, the performance of SA decreases and the difference between the two algorithms increases exponentially. Best performing HHMBO variant is 51%, 77%, 15%, 60%, and 210% better than the best performing HH variant for numbers of employees of 30, 50, 100, 150, and 200, respectively. The performance of HH is admissible for numbers of employees of 30 to 150, whereas when the number of employees increases to 200, HH performs exponentially worse.

For the experiment in which the number of employees is 500, the total iteration number determined with N^2 corresponds to 250,000. That is a reasonably big number of chances to search through the solution space. For this challenging case, the best performing HHMBO variant (RP_OI) is 274% better than SA, 138% better than HH, and 16% better than MBO.

HHMBO (RP_OI) obtained significantly better results in complex FOSSP instances. RP_OI applies each of the four problem specific heuristics consecutively due to the permutation order. Each heuristic is applied to an equal number of instances. Each heuristic has a particular ability to discover search space. Accomplishment of HHMBO (RP_OI) exposes that when the algorithm is given a great search ability, using heuristics equal numbers of times helps us to find the best results. HHMBO is successful on catching values close to global optimum. HHMBO provides diversification along the search space by applying a variety of heuristics alternately; that is why it is promising for solving combinatorial optimization problems.

The standard deviation results gained by FOSSP experiments are presented in Table 13 to demonstrate the robustness of the algorithms. It is observed that HHMBO-RP_OI is more robust than its rivals. We can say that the HHMBO variants produce slightly better standard deviation results than their HH variants. This can be explained by the design of the algorithms as follows. The HHMBO variants wander within the solution space by exploiting some prioritized (leader) solutions more than HH do, which inherently makes them more robust.

The average results of each algorithm are illustrated in Figure 3 for each number of employees. When the number of employees is equal to 30, all algorithms produce close results to each other. As the number of employees increases, the distinction between the algorithms involved in the experiment becomes more prominent. HHMBO-RP_OI produced the best results for the most challenging experiment, when the number of employees is equal to 500. HHMBO versions figure out lower (better) results when they are compared with their HH versions.

We have also applied t-test to check if the differences are statistically significant. The best performing HHMBO-RP_OI and all other algorithms in the experiment are tested and the statistical results obtained by two-tailed t-test are given in Table 14. We observed that the p value for pairwise t-tests among the best performing proposed variant HHMBO-RP_OI and the state-of-the-art algorithms is 0.004% on average. The results imply that HHMBO-RP_OI is indeed statistically different from other algorithms.

6.3. Results on Benchmark Problems in QAPLIB. After observing that HHMBO variants get very successful results for the FOSSP instances, we wanted to see their ability in converging optimal solutions on a widely used problem. We applied HHMBO on QAP, where MBO is firstly applied and tested. If N is the size of a QAP instance, we limited the run time with N^3 iterations. This corresponded to 0.01 s for a problem of size 12 and to 48 s for a problem of size 100 on the specified machine.

We have utilized 41 QAPLIB instances with different densities. Density corresponds to the percentage of nonzero entries in the flow matrix. The instance names, densities, size of the instances, best known solutions (BKS), and the percentage deviations of heuristic results from BKS are given in Table 15. Problem instances are listed from the sparse one to the dense one. Only the minimum costs obtained in 10 runs are considered, since the aim is to check the ability of HHMBO in finding BKS.

The results indicate that AS_IO variant of HHMBO is accomplished on catching promising solutions. HHMBO (AS_OI) is up to 14.6% better than MBO on converging to the BKS. According to the average percentage deviation gained from 41 instances, HHMBO (AS_OI) converges 7.5% to BKS, while MBO is 8.2% close to it. HHMBO (AS_OI) performs 0.7% better than MBO on average. Best performing variant of this experiment is adaptive searching Only Improvement HHMBO. Adaptive searching strategy exploits low-level heuristics depending on their performances. This adaptive feature ensures choosing the most beneficial low-level heuristic for each specific case during the run. It has been observed from the experiments conducted on QAPLIB benchmark problems that HHMBO consistently allows finding better results, albeit slightly.

7. Conclusion

In this study, we introduced a novel hybrid computational intelligence algorithm to the literature and the mathematical model of a shift scheduling problem of a manufacturing company defined by including the fairness perspective, which is typically ignored especially in manufacturing industry. The novel algorithm is designed by embedding hyperheuristic (HH) improvements in the migrating bird optimization (MBO) algorithm, called HHMBO. In this way, the exploitation capability of MBO is improved and the solution space is explored in a more diversified manner.

The problem that we tackled is called fairness oriented shift scheduling problem (FOSSP). The model of FOSSP is firstly verified with a solver, where the solver is capable of catching global optima with large execution time even for small instances. Therefore, exploitation of heuristic based techniques was an apparent need. In line with this need, we applied HHMBO together with simulated annealing (SA),

hyperheuristics (HH), and classical migrating bird optimization (MBO). We have conducted computational experiments in order to assess the performance of the algorithms on synthetic data. Results show the superiority of HHMBO-RP_OI and it is observed that it outperforms its rivals by 40% on average for FOSSP. Besides, it is seen that HHMBO-RP_OI definitely offers better solutions in large problem instances, which is a desired property in practice and gap in the literature. We believe that this good performance is due to integrating the exploitation capability of HH with MBO's exploration capability.

To justify the superiority of HHMBO over the other algorithms, we used QAP instances from the QAPLIB. This experiment was significant to understand whether the achievement of the algorithm is specific to the particular problem. We conducted computational experiments on 41 QAPLIB instances with assorted densities. According to the results of this experiment, AS_OI variant of the new hybrid algorithm could achieve up to 14.6% better results than MBO.

There are several interesting directions for future research. The approach may be applied on different types of optimization problems to highlight its wide applicability and generality. An example may be the test suites given in [52]. Furthermore, a comparative study may be conducted on swarm based hybrid algorithms. A multiobjective variation of HHMBO may be developed and may be applied on a set of multiobjective problems. Another research direction is to help solve employee transportation issues while constructing weekly schedules.

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.

Acknowledgments

This work was supported by the Marmara University Scientific Research Committee under the Project ID FEN-C-DRP-131217-0671.

References

[1] A. T. Ernst, H. Jiang, M. Krishnamoorthy, B. Owens, and D. Sier, "An annotated bibliography of personnel scheduling and rostering," *Annals of Operations Research*, vol. 127, no. 1–4, pp. 21–144, 2004.

[2] P. De Causmaecker, P. Demeester, G. V. Berghe, and B. Verbeke, "Analysis of real-world personnel scheduling problems," in *Proceedings of the 5th International Conference on Practice and Theory of Automated Timetabling*, pp. 183–197, Pittsburgh, PA, USA, August 2004.

[3] M. Defraeye and I. Van Nieuwenhuyse, "Staffing and scheduling under nonstationary demand for service: a literature review," *Omega*, vol. 58, pp. 4–25, 2016.

[4] M. Erhard, J. Schoenfelder, A. Fügener, and J. O. Brunner, "State of the art in physician scheduling," *European Journal of Operational Research*, vol. 265, no. 1, pp. 1–18, 2018.

[5] A. T. Ernst, H. Jiang, M. Krishnamoorthy, and D. Sier, "Staff scheduling and rostering: a review of applications, methods and models," *European Journal of Operational Research*, vol. 153, no. 1, pp. 3–27, 2004.

[6] J. Van den Bergh, J. Beliën, P. De Bruecker, E. Demeulemeester, and L. De Boeck, "Personnel scheduling: a literature review," *European Journal of Operational Research*, vol. 226, no. 3, pp. 367–385, 2013.

[7] P. Brucker, R. Qu, and E. Burke, "Personnel scheduling: models and complexity," *European Journal of Operational Research*, vol. 210, no. 3, pp. 467–473, 2011.

[8] K. R. Baker, "Workforce allocation in cyclical scheduling problems: a survey," *Journal of the Operational Research Society*, vol. 27, no. 1, pp. 155–167, 1976.

[9] H. K. Alfares, "Survey, categorization, and comparison of recent tour scheduling literature," *Annals of Operations Research*, vol. 127, no. 1–4, pp. 145–175, 2004.

[10] C. Seifi, M. Schulze, and J. Zimmermann, "A new mathematical formulation for a potash-mine shift scheduling problem with a simultaneous assignment of machines and workers," *European Journal of Operational Research*, vol. 292, no. 1, pp. 27–42, 2021.

[11] J. O. Brunner and R. Stolletz, "Stabilized branch and price with dynamic parameter updating for discontinuous tour scheduling," *Computers & Operations Research*, vol. 44, pp. 137–145, 2014.

[12] J. Atlason, M. A. Epelman, and S. G. Henderson, "Call center staffing with simulation and cutting plane methods," *Annals of Operations Research*, vol. 127, no. 1–4, pp. 333–358, 2004.

[13] J. G. Morris and M. J. Showalter, "Simple approaches to shift, days-off and tour scheduling problems," *Management Science*, vol. 29, no. 8, pp. 942–950, 1983.

[14] F. F. Easton and D. F. Rossin, "Equivalent alternate solutions for the tour scheduling problem," *Decision Sciences*, vol. 22, no. 5, pp. 985–1007, 1991.

[15] F. F. Easton and D. F. Rossin, "A stochastic goal program for employee scheduling," *Decision Sciences*, vol. 27, no. 3, pp. 541–568, 1996.

[16] R. Stolletz, "Operational workforce planning for check-in counters at airports," *Transportation Research Part E: Logistics and Transportation Review*, vol. 46, no. 3, pp. 414–425, 2010.

[17] H. K. Alfares, "An efficient two-phase algorithm for cyclic days-off scheduling," *Computers & Operations Research*, vol. 25, no. 11, pp. 913–923, 1998.

[18] D. S. Altner, E. K. Mason, and L. D. Servi, "Two-stage stochastic days-off scheduling of multi-skilled analysts with training options," *Journal of Combinatorial Optimization*, vol. 38, no. 1, pp. 111–129, 2019.

[19] H. C. Lau, "On the complexity of manpower shift scheduling," *Computers & Operations Research*, vol. 23, no. 1, pp. 93–102, 1996.

[20] G. Alp and A. F. Alkaya, "Improving the quality of personnel scheduling by incorporating fairness," *International Journal of Modeling and Optimization*, vol. 9, no. 2, pp. 97–101, 2019.

[21] A. Yurtkuran, B. Yagmahan, and E. Emel, "A novel artificial bee colony algorithm for the workforce scheduling and balancing problem in sub-assembly lines with limited buffers," *Applied Soft Computing*, vol. 73, pp. 767–782, 2018.

[22] M. Günther and V. Nissen, "Application of particle swarm optimization to the British telecom workforce scheduling problem," vol. 4, 2013.

[23] M. N. Janardhanan, Z. Li, and P. Nielsen, "Model and migrating birds optimization algorithm for two-sided assembly line worker assignment and balancing problem," *Soft Computing*, vol. 23, no. 21, pp. 11263–11276, 2019.

[24] H. Algethami and D. Landa-Silva, "Diversity-based adaptive genetic algorithm for a workforce scheduling and routing problem," in *Proceedings of the 2017 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1771–1778, IEEE, Donostia, San Sebastián, Spain, June 2017.

[25] H. Algethami, A. Martínez-Gavara, and D. Landa-Silva, "Adaptive multiple crossover genetic algorithm to solve workforce scheduling and routing problem," *Journal of Heuristics*, vol. 25, no. 4, pp. 753–792, 2019.

[26] W.-L. Liu, Y.-J. Gong, W.-N. Chen, Z. Liu, H. Wang, and J. Zhang, "Coordinated charging scheduling of electric vehicles: a mixed-variable differential evolution approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 12, pp. 5094–5109, 2019.

[27] K. Sethanan and R. Pitakaso, "Improved differential evolution algorithms for solving generalized assignment problem," *Expert Systems with Applications*, vol. 45, pp. 450–459, 2016.

[28] F. Zhao, L. Zhao, L. Wang, and H. Song, "An ensemble discrete differential evolution for the distributed blocking flowshop scheduling with minimizing makespan criterion," *Expert Systems with Applications*, vol. 160, Article ID 113678, 2020.

[29] S. Zhou, L. Xing, X. Zheng, N. Du, L. Wang, and Q. Zhang, "A self-adaptive differential evolution algorithm for scheduling a single batch-processing machine with arbitrary job sizes and release times," *IEEE transactions on cybernetics*, vol. 51, no. 3, 2019.

[30] E. K. Burke, G. Kendall, and E. Soubeiga, "A tabu-search hyperheuristic for timetabling and rostering," *Journal of Heuristics*, vol. 9, no. 6, pp. 451–470, 2003.

[31] S. Pan, M. Akplogan, N. Touati, L. Létocart, R. Wolfler Calvo, and L.-M. Rousseau, "A hybrid heuristic for the multi-activity tour scheduling problem," *Electronic Notes in Discrete Mathematics*, vol. 69, pp. 333–340, 2018.

[32] N. A. Hernández-Leandro, V. Boyer, M. A. Salazar-Aguilar, and L.-M. Rousseau, "A matheuristic based on Lagrangian relaxation for the multi-activity shift scheduling problem," *European Journal of Operational Research*, vol. 272, no. 3, pp. 859–867, 2019.

[33] F. Zhao, L. Zhang, Y. Zhang, W. Ma, C. Zhang, and H. Song, "A hybrid discrete water wave optimization algorithm for the no-idle flowshop scheduling problem with total tardiness criterion," *Expert Systems with Applications*, vol. 146, Article ID 113166, 2020.

[34] S. Asta, E. Özcan, and T. Curtois, "A tensor based hyper-heuristic for nurse rostering," *Knowledge-Based Systems*, vol. 98, pp. 185–199, 2016.

[35] H. Li, Y. Wang, S. Li, and S. Li, "A column generation based hyper-heuristic to the bus driver scheduling problem," *Discrete Dynamics in Nature and Society*, vol. 201510 pages, 2015.

[36] F. Garza-Santisteban, R. Sánchez-Pámanes, L. A. Puente-Rodríguez et al., "A simulated annealing hyper-heuristic for job shop scheduling problems," in *Proceedings of the 2019 IEEE Congress on Evolutionary Computation (CEC)*, pp. 57–64, IEEE, Wellington, New Zealand, June 2019.

[37] M. Alinia Ahandani, M. T. Vakil Baghmisheh, M. A. Badamchi Zadeh, and S. Ghaemi, "Hybrid particle swarm optimization transplanted into a hyper-heuristic structure for solving examination timetabling problem," *Swarm and Evolutionary Computation*, vol. 7, pp. 21–34, 2012.

[38] L. Chen, H. Zheng, D. Zheng, and D. Li, "An ant colony optimization-based hyper-heuristic with genetic programming approach for a hybrid flow shop scheduling problem," in *Proceedings of the 2015 IEEE Congress on Evolutionary Computation (CEC)*, pp. 814–821, IEEE, Sendai, Japan, May 2015.

[39] V. Pandiri and A. Singh, "A hyper-heuristic based artificial bee colony algorithm for k-interconnected multi-depot multi-traveling salesman problem," *Information Sciences*, vol. 463-464, pp. 261–281, 2018.

[40] E. Duman, M. Uysal, and A. F. Alkaya, "Migrating birds optimization: a new metaheuristic approach and its performance on quadratic assignment problem," *Information Sciences*, vol. 217, pp. 65–77, 2012.

[41] W. Jang, H. H. Lim, T. J. Crowe, G. Raskin, and T. E. Perkins, "The Missouri lottery optimizes its scheduling and routing to improve efficiency and balance," *Interfaces*, vol. 36, no. 4, pp. 302–313, 2006.

[42] W. Y. Szeto and H. K. Lo, "Transportation network improvement and tolling strategies: the issue of intergeneration equity," *Transportation Research Part A: Policy and Practice*, vol. 40, no. 3, pp. 227–243, 2006.

[43] E. K. Burke, M. Gendreau, M. Hyde et al., "Hyper-heuristics: a survey of the state of the art," *Journal of the Operational Research Society*, vol. 64, no. 12, pp. 1695–1724, 2013.

[44] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[45] D. Öz and I. Öz, "Scalable parallel implementation of migrating birds optimization for the multi-objective task allocation problem," *The Journal of Supercomputing*, vol. 77, no. 3, pp. 2689–2712, 2021.

[46] Y. Han, J.-Q. Li, D. Gong, and H. Sang, "Multi-objective migrating birds optimization algorithm for stochastic lot-streaming flow shop scheduling with blocking," *IEEE Access*, vol. 7, pp. 5946–5962, 2018.

[47] T. Meng, Q.-K. Pan, J.-Q. Li, and H.-Y. Sang, "An improved migrating birds optimization for an integrated lot-streaming flow shop scheduling problem," *Swarm and Evolutionary Computation*, vol. 38, pp. 64–78, 2018.

[48] A. Sioud and C. Gagné, "Enhanced migrating birds optimization algorithm for the permutation flow shop problem with sequence dependent setup times," *European Journal of Operational Research*, vol. 264, no. 1, pp. 66–73, 2018.

[49] Z. Zhang, Q. Tang, D. Han, and Z. Li, "Enhanced migrating birds optimization algorithm for u-shaped assembly line balancing problems with workers assignment," *Neural Computing & Applications*, vol. 31, no. 11, pp. 7501–7515, 2019.

[50] A. F. Alkaya, R. Algin, Y. Sahin, M. Agaoglu, and V. Aksakalli, "Performance of migrating birds optimization algorithm on continuous functions," in *Proceedings of the International Conference in Swarm Intelligence*, pp. 452–459, Springer, Hefei, China, October 2014.

[51] R. E. Burkard, S. E. Karisch, and F. Rendl, "Qaplib–a quadratic assignment problem library," *Journal of Global Optimization*, vol. 10, no. 4, pp. 391–403, 1997.

[52] G. Wu, R. Mallipeddi, and P. N. Suganthan, "Problem definitions and evaluation criteria for the CEC 2017 competition on constrained real-parameter optimization," Technical Report, National University of Defense Technology, Changsha, Hunan, PR China and Kyungpook National University, Daegu, South Korea and Nanyang Technological University, Singapore, 2017.