Research Article

Multiobjective Sustainable Order Allocation Problem Optimization with Improved Genetic Algorithm Using Priority Encoding

Zhiqiang Fan ¹, Shanshan Li ², and Zhijun Gao³

¹School of Business Administration & Research Center of Energy Economy, Henan Polytechnic University, Jiaozuo 454000, China

²School of Finance and Economics, Henan Polytechnic University, Jiaozuo 454000, China

³School of Transport & Communications, Shanghai Maritime University, Shanghai 201306, China

Correspondence should be addressed to Shanshan Li; lss925@hpu.edu.cn

Received 12 August 2019; Revised 7 November 2019; Accepted 10 November 2019; Published 26 November 2019

Academic Editor: Florin Stoican

Copyright © 2019 Zhiqiang Fan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recently, incorporating carbon emissions into order allocation decisions has attracted considerable attention among scholars and industrialists. Moreover, affected by the random fluctuations of the man, machine, material, method, and environment (4M1E), the production process is usually imperfect with defective products. Reducing product defective rates can effectively improve the quality of the order allocation process. Therefore, considering product defective rate and carbon emission, a multiobjective integer nonlinear programming (INLP) formulation is presented to address this multiproduct, multiperiod, and multi-OEM order allocation problem. Furthermore, exploring the existing literatures, an improved genetic algorithm using priority encoding (IGAUPPE) is put forward as a novel optimization technique. Finally, numerical experiments are conducted to validate the correctness of the proposed INLP model as well as the effectiveness of the proposed algorithm. Compared with the genetic algorithm using binary encoding (GAUBE), genetic algorithm using two-layer encoding (GAUTE), and LINGO software, the experiment results show that IGAUPPE can improve the efficiency and effectiveness within the predetermined time limit when solving large-scale instances.

1. Introduction

More than a decade ago, the diversified needs of consumers contributed to changes in the production system. The original make to stock (MTS) system gradually transformed into the make to order (MTO) system in many production industries. In the MTO supply chain, order receiving and allocation operation becomes one of the most important tasks, which is the primary data source for the entire MTO supply chain. In general, order allocation problem (OAP) aims at allocating received customers' orders to various manufacturers for production. In the process of order allocation, the original equipment manufacturers (OEMs) can be commonly assessed in the light of multiple standards, which contain the specific price, capacity, quality, and efficiency [1]. Moreover, order quantities are allocated to

OEMs by decision-makers in line with their requirements to reduce cost [2].

Recently, new changes of consumer demand have taken place. Because carbon emission is relevant to many environmental problems, such as global warming and extreme climate, more and more consumers are paying attention to environment-friendly products. A survey by Bull [3] showed that consumers are willing to buy low-carbon products at a higher price. As a result, many OEMs invest on carbon emission reduction facilities and technologies to manufacture low-carbon products. Even some companies are beginning to label the carbon emissions during production on the products, such as H&M and Levis [4].

On the contrary, affected by the random fluctuations of the man, machine, material, method, and environment (4M1E), the production process is imperfect and produces

usually manufacture-defective products. It is also unavoidable to manufacture defective products when using these newly invested facilities and technologies.

Motivated by these practical challenges, this paper develops a three-echelon supply chain consisting of one original brand manufacturer (OBM), several OEMs, and various retailers with multiproduct and multiperiod. From the perspective of decision-makers, we study the order allocation problem in order to maximize overall benefits. To the best of our knowledge, the paper mainly has the following contributions. First, the study fills the research gap on the comprehensive consideration of carbon emission, product defective, quantity discounts, and multiperiod. Second, an improved genetic algorithm using priority encoding is put forward to improve the efficiency and effectiveness of the solving process.

The rest of this paper is organized as follows: Section 2 gives a detailed review on related literatures. Section 3 introduces the model used in this study. The proposed IGAUPE procedure is elaborated in Section 4. Section 5 validates the proposed INLP model and IGAUPE. Section 6 concludes the paper and future research direction.

2. Literature Review

2.1. Order Allocation. Proper order allocation should be a good trade off between efficiency and effectiveness in the supply chain. Most articles took traditional criteria (e.g., cost, time, price, and profit) as objectives to optimize the order allocation problem. For minimizing total time of all orders, Guo et al. [5] presented a single-product and single-plant order allocation formulation, and a genetic algorithm using order encoding (GAUOE) was developed to solve the proposed model. Extending the previous study, Guo et al. [6] introduced a hybrid intelligent optimization model to analyze the multiproduct and multiplant order allocation planning problem, which coordinates order tardiness, throughput time, and idle time simultaneously. Different from the above studies, Seifbarghy and Esfandiari [7] introduced a novel model of multisupplier and multiproduct order allocation with transaction costs over multiperiod. In their research, the genetic algorithm using binary encoding (GAUBE) and the standard simulated annealing (SA) were presented to solve the proposed mixed-integer programming (MIP) formulation. Furthermore, Nazari-Shirkouhi et al. [8] formulated a multiobjective linear programming model for multiproduct and multisupplier, which devoted to minimizing cost, number of rejected items, and late-delivered items. Based on the load equilibrium among suppliers, Renna and Perrone [9] proposed an order allocation policy. Using LINGO software, Gupta et al. [10] proposed a fuzzy multiobjective programming model to minimize the transportation cost and delivery time.

In particular, in view that quantity discount pricing can save cost among suppliers and buyers, some literatures had integrated it into the order allocation problem. For example, Jain et al. [11] took into account all-unit discount policy, incremental-unit discount, and their combination. They developed three heuristic methods to find the optimal order

quantity, i.e., genetic algorithm, artificial bee colony, and chaotic bee colony. Duan and Ventura [12] formulated a multiperiod mixed-integer linear programming (MILP) model with single product and multisupplier; meanwhile, lead time and order frequency are took into account in the model. Alegoz and Yapicioglu [13] used TOPSIS and AHP to determine the weights of different objectives.

Moreover, some researchers studied the order allocation problem under disruption risk. Hosseini et al. [14] developed a stochastic biobjective MIP model to demonstrate resilient supplier selection and order allocation problem. A biobjective mixed-integer linear programming (MILP) model was proposed by Goh and PrasannaVenkatesan [15] and was solved by AHP, PROMETHEE, and particle swarm algorithm.

Furthermore, some authors extended the research to logistic service supply chain. For profit maximizing, Liu et al. [27] studied the order allocation problem in logistic service supply chain. For cost-efficiency, Gao et al. [28] presented a multiobjective model of one logistic service integrator and multiple functional service providers. The model is solved by LINGO software. Li [16] proposed a genetic algorithm using two-layer encoding (GAUTE) consisting of binary and real numbers to solve the model in [28]. Under the mass customization of logistics services, Hu et al. [17] introduced a multiobjective mixed-integer nonlinear programming (MINLP) model to maximize the profit and overall satisfaction, and the genetic algorithm using two-layer encoding (GAUTE) consisting of binary and real numbers is also developed.

Recently, there are a number of literatures referring to sustainable criteria (e.g., carbon emission in production and logistics). For example, Gupta et al. [18] presented a multiobjective integer linear programming model under three different scenarios, i.e., the best scenario, the likeliest scenario, and the worst scenario. Considering carbon emission, Govindan and Sivakumar [19] proposed a multiobjective linear programming model solved by LINGO software. From the perspective of green criteria, Babbar and Amin [21] explored the supplier selection and order allocation using a two-stage methodology. For minimizing cost and maximizing the value of purchasing, Hamdan and Cheaitou [22] introduced a biobjective integer nonlinear programming (INLP) model to discuss the order allocation problem and a three-stage integer linear programming (ILP) model to research all-unit quantity discounts and varying supplier availability in Cheaitou and Hamdan [23]. Combining environmental and economic factors, Ghorabae et al. [20] formulated a multiobjective linear programming model solved by a fuzzy programming approach. Taking sustainability into account, Ghadimi et al. [24] presented two submodels for supplier selection and order allocation in the process of decision-making. To be specific, the first submodel dealt with supplier selection by the fuzzy inference system; for the second submodel, the optimal order quantities of each supplier is obtained by minimizing purchasing cost and maximizing supplier's sustainability performance. Considering multiobjectives, i.e., cost, carbon emission, and social responsibility, Moheb-Alizadeh and Handfield [25]

presented a ε -constraint method and Benders decomposition algorithm to solve the order allocation problem. By incorporating environmental impact and social impact, Mohammed et al. [26] developed a multiobjective model to allocate the optimal order quantities with single product and multisupplier. Table 1 gives a summary and comparison of the previous literatures related to the order allocation problem. Obviously, these papers have explored the order allocation problem using different methodologies, which enriched the current literature and provided a solid foundation for further research studies in our study.

However, to the best of our knowledge, few studies considered product quality, carbon emission, and quantity discounts at the same time. Although Gupta et al. [18] considered these three factors simultaneously, their research was assumed in the single period, which simplified the complexity of OAP in real-world environments.

2.2. Optimization Techniques. Order allocation belongs to a combinatorial optimization problem [6]. Many techniques, including classical optimization techniques and intelligent optimization techniques, had been utilized to obtain the optimal or near-optimal solutions. The existing optimization techniques for OAP are as follows: simulated annealing [7], particle swarm optimization [15], chaotic bee colony [11], hybrid algorithm [6, 25], genetic algorithm [5, 7, 16, 17], and LINGO software [10, 18, 19, 28, 29].

However, these optimization techniques still have some drawbacks and great improvement rooms (see Table 2). LINGO is undoubtedly the most popular technique to seek the optimal solutions, but it cannot be used to effectively solve large-scale OAP due to the complexity of the problem. For the large-scale OAP, the genetic algorithm is evidently the most commonly used due to their heuristic nature. Although the genetic algorithm is really suitable to solve combinatorial optimization problems and NP-hard problems, to some extent, the encoding rules used in the existing literatures reduce the efficiency of the GA. The four papers developed three encoding rules as follows: order-based encoding [5], binary encoding [7], and two-layer encoding consisting of binary and real numbers [16, 17]. The genetic algorithm using order-based encoding (GAUOE) rule developed by Guo et al. [5] is suitable for single sourcing OAP, in which one order is only assigned to one supplier. Because we attempt to study the multiply sourcing OAP in which one order can be fulfilled by multiple suppliers, the order-based encoding rule of Guo et al. [5] cannot be used in this paper. The latter two encoding rules, i.e., binary encoding [7] and two-layer encoding [16, 17], are suitable for multisourcing OAP. However, based on the latter two encoding rules, the GA will generate a large number of illegal solutions after the crossover and mutation operations. Repair of illegal solutions will increase the running time of GA and reduce the efficiency of GA.

This paper aims at filling the research gap and improving the efficiency and effectiveness of GA. The main contributions of the paper are as follows:

- (i) In view that few literatures took into account product quality, carbon emissions, quantity discounts, and multiperiod at the same time, we extend the existing model by considering defective rate, carbon emissions, quantity discounts, and multiperiod simultaneously. A novel integer nonlinear programming (INLP) model is presented to address this multiproduct and multi-OEM order allocation problem.
- (ii) We propose an improved GA using priority encoding (IGAUBE) to improve the solving efficiency. Compared with the GAUBE, GAUTE, and LINGO software, the performance evaluation of IGAUBE reveals that IGAUBE improves the efficiency and effectiveness of the solving process within the predetermined time limit when solving large-scale instances.

3. The Model

3.1. Problem Description. In order to reasonably allocate the order tasks to OEMs within the planning period, this paper will study the order allocation problem considering product defective, carbon emission, and quantity discounts. In addition, this study aims at determining the production quantity and production time for each OEM, so as to minimize the transaction costs, purchasing costs, number of defective products, number of late delivery products, and carbon emission of all orders.

Figure 1 illustrates the sketch map of the supply chain structure in the study, which consists of a single original brand manufacturer (OBM), several OEMs, and various retailers. Considering a multiperiod planning horizon, the OBM accepts and aggregates orders of all retailers so as to obtain the total product demand in each period. The decision-maker considers the production capacity, transaction cost, product price, defective rate, and unit carbon emission. In addition, he arranges the order plans of each OEM's product in each period. Considering various influencing factors in the process of order allocation, we pursue the minimum of transaction costs, purchasing costs, number of defective products, number of late-delivered products, and carbon emissions of all orders.

Assuming that the planning horizon can be divided into multiple periods, each OEM can produce a variety of homogeneous products. It is worthwhile to mention that products in each OEM have a distinct defective rate. Relevant parameters used in this paper are shown in Table 3.

3.2. Multiobjective Model. The multiobjective model for the order allocation problem can be formulated as

$$\text{Min}F_1 = \sum_j \sum_i \sum_t \sum_r a_{ij} y_{ijtr}, \quad (1)$$

$$\text{Min}F_2 = \sum_j \sum_i \sum_t \sum_r (1 - d_{ijtr}) A_{ijt} x_{ijtr}, \quad (2)$$

TABLE 1: Literature review summary.

Paper	Objective			Quantity discount	Product		Period		Supplier	
	Tradition	Quality	Carbon emission		Single	Multi	Single	Multi	Single	Multi
[5]	✓				✓		✓		✓	
[6]	✓			✓		✓	✓			✓
[7]	✓					✓		✓		✓
[8]	✓					✓	✓			✓
[9, 14]	✓				✓			✓		✓
[10]	✓				✓		✓			✓
[11]	✓			✓	✓		✓			✓
[12]	✓			✓	✓			✓		✓
[13]	✓			✓		✓	✓			✓
[16, 17]	✓					✓		✓		✓
[18]	✓	✓	✓	✓		✓	✓			✓
[19, 20]	✓	✓	✓		✓		✓			✓
[21]	✓	✓	✓			✓		✓		✓
[22]	✓		✓		✓			✓		✓
[23]	✓		✓	✓	✓			✓		✓
[24]	✓	✓	✓			✓	✓			✓
[25]	✓		✓			✓		✓		✓
[26]	✓		✓		✓		✓			✓
This study	✓	✓	✓	✓		✓		✓		✓

TABLE 2: Differences between this paper and relevant studies.

Literature	[10, 18, 19, 28, 29]	[5]	[7]	[16, 17]	This study
Optimization technique	LINGO	GAUOE	GAUBE	GAUTE	IGAUBE
Can the technique solve large-scale OAP problem?	No	Yes	Yes	Yes	Yes
Can the technique solve multiply sourcing OAP?	Yes	No	Yes	Yes	Yes
Chromosome encoding rule	—	Order-based	Binary	Two layers consisting of binary and real numbers	Priority
Can the technique avoid illegal solutions after crossover and mutation in multisourcing OAP?	—	—	No	No	Yes

FIGURE 1: The supply chain structure in the study.

$$\text{Min}F_3 = \sum_j \sum_i \sum_t \sum_r p_{ij} x_{ijtr}, \quad (3)$$

$$\text{Min}F_4 = \sum_j \sum_i \sum_t \sum_r t_{ij} x_{ijtr}, \quad (4)$$

$$\text{Min}F_5 = \sum_j \sum_i \sum_t \sum_r u_{ij} x_{ijtr}, \quad (5)$$

where equation (1) minimizes the total transaction costs during the planning period, equation (2) minimizes the total purchasing costs, equation (3) minimizes the total number of defective products, equation (4) minimizes the total number of late-delivery products, and equation (5) minimizes carbon emission of all orders.

Obviously, for the above functions, it is hard to obtain the optimal solution to satisfy every objective simultaneously. Meanwhile, the final order allocation solution should be a concrete scheduling that can be carried out directly and quickly, rather than a huge Pareto non-dominated solution sets. In this study, the classical objective weighting method is employed to represent corresponding weights for each objective and to convert multiobjectives into single objectives:

TABLE 3: Notations for the model.

Indices	
i	Index of OEM, $i = 1, 2, \dots, I$
j	Index of product, $j = 1, 2, \dots, J$
t	Index of period, $t = 1, 2, \dots, T$
r	Index of discount interval, $r = 1, 2, \dots, R$
Parameters	
A_{ijt}	Original price of product j supplied by OEM i in period t
d_{ijtr}	Price discount rate of discount interval r when product j is supplied by OEM i in period t
$Q_{ijtr}^{\max}$	Upper limit quantity of discount interval r when product j is supplied by OEM i in period t
$Q_{ijtr}^{\min}$	Lower limit quantity of discount interval r when product j is supplied by OEM i in period t
n_{ij}	Number of discount intervals of product j for OEM i
u_{ij}	Unit carbon emission of product j manufactured by OEM i
p_{ij}	Defective rate when OEM i produces product j
t_{ij}	Late delivery rate of product j supplied by OEM i
P_j	Maximum acceptable defective rate of product j within the planning horizon
T_j	Maximum acceptable late delivery rate of product j within the planning horizon
D_{jt}	Demand for product j in period t
C_{ijt}	Capacity of OEM i to manufacture product j in period t
a_{ij}	Transaction cost of purchasing product j from OEM i
Decision variables	
x_{ijtr}	Order quantity in the discount interval r when product j is supplied by OEM i in period t
y_{ijtr}	0-1 variable, $y_{ijtr} = 1$, if product j is supplied by OEM i in period t within the discount interval r ; otherwise, $y_{ijtr} = 0$

$$\text{Min}F = \lambda_1 F_1 + \lambda_2 F_2 + \lambda_3 F_3 + \lambda_4 F_4 + \lambda_5 F_5, \quad (6)$$

where $\lambda_1, \lambda_2, \lambda_3, \lambda_4$, and λ_5 represents the weight of five subobjectives, respectively, and their sum equals 1.

Subject to

$$\sum_i \sum_r y_{ijtr} x_{ijtr} \geq D_{jt}, \quad \forall j, t, \quad (7)$$

$$\sum_i \sum_t \sum_r p_{ij} x_{ijtr} \leq P_j \sum_t D_{jt}, \quad \forall j, \quad (8)$$

$$\sum_i \sum_t t_{ij} x_{ijtr} \leq T_j \sum_t D_{jt}, \quad \forall j, \quad (9)$$

$$x_{ijtr} \leq C_{ijt} y_{ijtr}, \quad \forall i, j, t, r, \quad (10)$$

$$\sum_r y_{ijtr} \leq 1, \quad \forall i, j, t, \quad (11)$$

$$y_{ijtr} Q_{ijtr}^{\min} \leq x_{ijtr} < y_{ijtr} Q_{ijtr}^{\max}, \quad \forall i, j, t, \forall r < n_{ij}, \quad (12)$$

$$y_{ijtr} Q_{ijtr}^{\min} \leq x_{ijtr} \leq y_{ijtr} Q_{ijtr}^{\max}, \quad \forall i, j, t, \forall r = n_{ij}, \quad (13)$$

$$Q_{ijtr}^{\max} = C_{ijt}, \quad \forall i, j, t, \forall r = n_{ij}, \quad (14)$$

$$Q_{ijtr}^{\min} = Q_{ijtr-1}^{\max}, \quad \forall i, j, t, \forall r > 1, \quad (15)$$

$$Q_{ijtr}^{\min} = 0, \quad \forall i, j, t, \forall r = 1, \quad (16)$$

$$x_{ijtr} \geq 0 \text{ and integer}, \quad \forall i, j, t, r, \quad (17)$$

$$y_{ijtr} \in \{0, 1\}, \quad \forall i, j, t, r, \quad (18)$$

Equation (7) represents the demand constraint where the order demand must be satisfied. Equation (8) represents the maximum acceptable defective rate constraint. Equation (9) represents the maximum acceptable late delivery rate constraint. Equation (10) represents the OEM production capacity constraint. Equation (11) guarantees that mostly one discount interval is selected for each OEM. Equations (12)–(16) ensure the product quantity of each OEM is between any two consecutive breakpoints. Equation (17) expresses the integrality of decision variable x_{ijtr} . Equation (18) means that y_{ijtr} is binary variables.

4. Algorithm Design

Multiobjective order allocation has been proved to be a NP-hard problem [7]. Genetic algorithm (GA), dealing simultaneously with a set of population, is well apt to solve the NP-hard problem. We have proposed an improved genetic algorithm using priority encoding (IGAUPPE) to optimize objective function F given by Equation (6) and constrains (7)–(18). The specific process is shown in Figure 2.

4.1. Encoding and Decoding. Encoding is the most key issue in the genetic algorithm. An appropriate chromosome encoding will facilitate all the subsequent operations of the GA and improve efficiency. In previous literatures, there are three encoding rules of GA for the order allocation problem: order-based encoding [5], binary encoding [7], and two-layer encoding consisting of binary and real numbers [16, 17]. The order-based encoding rule is not suitable for this paper because our order allocation is not a single-sourcing problem. Based on the latter two encoding rules, they need many repair mechanisms to get feasible solutions after crossover and mutation. And these repair mechanisms will spend a lot of time in algorithms. In this study, we use priority encoding to avoid these repair mechanisms and improve efficiency of the genetic algorithm. The priority encoding of the genetic algorithm has successfully applied for many engineering problems, such as supply chain network optimization, project scheduling, shortest path, and traveling salesman problem.

The priority encoding rule is introduced to multiple OEMs, multiple products, and multiple periods order allocation problem. Each chromosome consists of T gene segments. The t th segment of chromosome represents the

FIGURE 2: Specific process of IGAUPE.

solution in period t . In each segment, the OEM id is represented by the position of gene, and the order priority among candidates is represented by the value of gene. Figure 3 shows an example of the chromosome representation, in which there are four OEMs, two products, and two periods. In period 1 ($t = 1$), for the product 1 ($j = 1$), the order allocation sequence is OEM 4 $\rightarrow$ OEM 3 $\rightarrow$ OEM 2 $\rightarrow$ OEM 1; for the product 2 ($j = 2$), the order allocation sequence is OEM 4 $\rightarrow$ OEM 1 $\rightarrow$ OEM 3 $\rightarrow$ OEM 2. In period 2 ($t = 2$), for the product 1 ($j = 1$), the order allocation sequence is OEM 2 $\rightarrow$ OEM 4 $\rightarrow$ OEM 1 $\rightarrow$ OEM 3; for the product 2 ($j = 2$), the order allocation sequence is OEM 3 $\rightarrow$ OEM 1 $\rightarrow$ OEM 4 $\rightarrow$ OEM 2.

The chromosome of IGAUPE is decoded in units of segments on the forward direction. Algorithm 1 describes the pseudocode of chromosome segment decoding. Taking the first segment of the chromosome in Figure 3 as an example, the decoding process is shown in Table 4.

4.2. Fitness Function. For a chromosome, its fitness value reflects that how good it is and decides whether it is selected in the next generation. It is noteworthy that we will choose a chromosome with a higher fitness value. In order to clearly reflect the subtle changes in the fitness value, we set the numeric value of the numerator to 10^8 . The fitness function in the IGAUPE is as follows:

$$f = \frac{10^8}{F}, \quad (19)$$

where F represents the objective function value given by equation (6).

FIGURE 3: Chromosome representation of IGAUPE.

4.3. Selection. A combined method of hybrid roulette and elite reservation is adopted as the selection strategy in this paper. The best individual in the parent population is directly retained to the next population. The other individuals in the next population are chosen by the roulette method. The roulette method is also applied to select parents for the crossover operator and mutation operator.

4.4. Crossover. Crossover operation is the fundamental way to generate new chromosomes. According to the characteristic of the chromosome structure, we take the segment of chromosome as the unit to operate uniform crossover. The crossover operation is shown in Figure 4.

4.5. Mutation. The mutation operation can effectively improve the search efficiency of the genetic algorithm and maintain the diversity of population. Mutation is performed by a swap operator. In the corresponding segment, we randomly select two genes and exchange their places. The mutation operation used in the proposed genetic algorithm is shown in Figure 5.

4.6. Stopping Rule. The common stopping rules of the genetic algorithm are as follows: (1) reaches the generation limit, (2) reaches the time limit, and (3) performance cannot

Input: I : set of OEMs; J : set of products
 D_j : demand for product j , $\forall j \in J$
 C_{ij} : capacity of OEM i to manufacture product j , $\forall i \in I, \forall j \in J$
 $v(i \times j)$: chromosome, $\forall i \in I, \forall j \in J$

Output: x_{ij} : quantity of product j supplied by OEM i , $\forall i \in I, \forall j \in J$

- (1) $x_{ij} \leftarrow 0, \forall i \in I, \forall j \in J$
- (2) Select the gene k with the highest priority value
- (3) $l \leftarrow \arg \max \{v(k), k \in I \times J\}$
 $// \text{ceil}(a)$ is a function that rounds the element of a to the nearest integer greater than or equal to a
- (4) $j^* = \text{ceil}(l/I)$, determine the product type
- (5) $i^* = l - (j^* - 1) \times I$, determine the OEM id
- (6) $x_{i^*j^*} = \min(D_{j^*}, C_{i^*j^*})$, allocate available product quantities
 $//$ Update availabilities on demand (D_{j^*}) and capacity ($C_{i^*j^*}$)
- (7) $D_{j^*} = D_{j^*} - x_{i^*j^*}; C_{i^*j^*} = C_{i^*j^*} - x_{i^*j^*}$
- (8) **if** $D_{j^*} = 0$, **then**
- (9)   **for** $s = I \times (j^* - 1) + 1$ to $I \times j^*$ **do**
- (10) $v(s) = 0$
- (11)   **end for**
- (12) **else**
- (13) $v(l) = 0$
- (14) **end if**
- (15) **for all** $i \in I, \forall j \in J$ **do**
- (16)   **if** each $v(i \times j) = 0$, **then**
- (17)   stop
- (18)   **else**
- (19)   go to step 1
- (20)   **end if**
- (21) **end for**

ALGORITHM 1: Pseudocode of chromosome segment decoding.

be improved. The first stopping rule is adopted for an illustrative instance in Section 5.1, and the second stopping rule is adopted for large-scale computational experiments in Section 5.2.

Algorithm 2 describes the pseudocode of the proposed IGAUPE.

5. Numerical Experiments

The proposed GA is programmed via Matlab software and tested on an Intel Core i5-2500 CPU @3.30 GHz with 4 GB RAM. Firstly, an illustrative instance is conducted to test the effectiveness of the INLP model and IGAUPE proposed in this study. Then, the performance of IGAUPE is tested compared with GAUBE, GAUTE, and LINGO software through large-scale instance sets. In the GAUBE, binary encoding developed by Seifbarghy and Esfandiari [7] is used for chromosome representation of the order allocation problem. In the GAUTE, two-layer encoding consisting of binary and real numbers developed by Li [16] and Hu et al. [17] is used for chromosome representation of the order allocation problem.

5.1. An Illustrative Instance. Let us consider an instance consisting of two OEMs, three products, three periods, and two discount intervals. The maximum acceptable defective rates of three products are 0.004, 0.006, and 0.008, respectively. The maximum acceptable late delivery rates of three products are 0.085, 0.12, and 0.009, respectively. The

FIGURE 4: Sketch of crossover operation.

FIGURE 5: Sketch of mutation operation.

price discount rates of two discount intervals are 0 and 0.1 in turns. The five weights are the same and equal to 0.2. The lower limit quantity of the second discount interval is 50% of the production capacity. The values of original price, capacity, and demand are shown in Table 5, and Table 6 shows the values of defective date, late delivery rate, unit carbon emission, and transaction cost.

Based on the minimal objective function value, the suitable values of parameters for the IGAUPE are gained

TABLE 4: Decoding process of the chromosome segment.

Iteration	$v(i \times j)$	D_j	C_{ij}	j^*	i^*	$x_{i^*j^*}$
0	[2 3 6 8 5 1 4 7]	(100,80)	[(30,20), (10,20), (30,40), (40,40)]	1	4	40
1	[2 3 6 0 5 1 4 7]	(60,80)	[(30,20), (10,20), (30,40), (0,40)]	2	4	40
2	[2 3 6 0 5 1 4 0]	(60,40)	[(30,20), (10,20), (30,40), (0,0)]	1	3	30
3	[2 3 0 0 5 1 4 0]	(30,40)	[(30,20), (10,20), (0,40), (0,0)]	2	1	20
4	[2 3 0 0 0 1 4 0]	(30,20)	[(30,0), (10,20), (0,40), (0,0)]	2	3	20
5	[2 3 0 0 0 0 0 0]	(30,0)	[(30,0), (10,20), (0,20), (0,0)]	1	2	10
6	[2 0 0 0 0 0 0 0]	(20,0)	[(30,0), (0,20), (0,20), (0,0)]	1	1	20
7	[0 0 0 0 0 0 0 0]	(0,0)	[(10,0), (0,20), (0,20), (0,0)]			

Input: Pop , P_c , and P_m .
Output: the best solution X

- (1) Generate a initial population (Pop)
- (2) $X \leftarrow$ the best solution in the initial Pop
- (3) **while** *stopping rule* is not met **do**
- (4)   **for** each solution in Pop **do**
- (5)   *Fitness* (Pop)
- (6)   **end for**
- (7) $X^* \leftarrow$ the best solution in the current Pop
- (8)   **if** X^* is better than X , **then**
- (9) $X \leftarrow X^*$
- (10)   **end if**
- // Crossover
- // $u01$ is a random number between 0 and 1; P_c is a crossover probability
- // p_1 , p_2 are two different parent individuals; C is a child solution
- (11)   **if** $u01 < P_c$, **then**
- (12) $p_1 \leftarrow$ selection (Pop);
- (13) $p_2 \leftarrow$ selection (Pop);
- (14) $C \leftarrow$ crossover (p_1 , p_2)
- (15)   **end if**
- // Mutation
- // P_m is a mutation probability; p_3 is a parent individual
- (16)   **if** $u01 < P_m$, **then**
- (17) $p_3 \leftarrow$ selection (Pop);
- (18) $C \leftarrow$ mutation (p_3)
- (19)   **end if**
- (20)   **Update the population**
- (21) **end while**

ALGORITHM 2: Pseudocode of IGAUPE.

from different values. They are set as follows: population size is 100, crossover probability is 0.8, mutation probability is 0.1, and generation is 1000. The objective function value is 227917.4, and the solutions of decision variables are listed in Table 7. Obviously, the proposed model can effectively simulate the decision-making process of order allocation, and the proposed IGAUPE is suitable to solve the model.

5.2. Large-Scale Computational Experiments. In this section, the effectiveness and efficiency of the proposed IGAUPE is validated by 10 sets. Each set consists of 10 examples; therefore, 100 examples are analyzed. These instances are solved by LINGO 13.0 to obtain the optimal or the current best solutions within 120 minutes. The solutions of LINGO are used to evaluate the performance of IGAUPE, GAUBE, and GAUTE. Table 8 lists the related parameters, which

contain the number of OEMs, products, periods, and intervals in each set. In each instance, the values of parameters are randomly generated within the value range shown in Table 9. The weight of each objective function remains 0.2; however, the set volume is increasing in turn.

Through equation (20), the performance of IGAUPE is compared with GAUBE, GAUTE, and LINGO 13.0 software. In the IGAUPE, the initial population is randomly generated. The values of parameters for the IGAUPE, GAUBE, and GAUTE are set as follows: population size is 100, crossover probability is 0.8, and mutation probability is 0.1. For comparison, a very large number for generations is set to ensure that the computation time of the IGAUPE, GAUBE, and GAUTE is more than 10 minutes. In other words, when the computation time of the IGAUPE, GAUBE, and GAUTE reaches 10 minutes, they stop running and will output the current best results:

TABLE 5: Values of A_{ijt} , C_{ijt} , and D_{jt} .

A_{ijt}		$i = 1$	$i = 2$	C_{ijt}		$i = 1$	$i = 2$	D_{jt}		
$j = 1$	$t = 1$	12	11	$j = 1$	$t = 1$	2000	1100	$j = 1$	$t = 1$	2000
	$t = 2$	15	13		$t = 2$	700	800		$t = 2$	1400
	$t = 3$	18	17		$t = 3$	1000	800		$t = 3$	1000
$j = 2$	$t = 1$	32	25	$j = 2$	$t = 1$	850	1000	$j = 2$	$t = 1$	1500
	$t = 2$	40	35		$t = 2$	800	1600		$t = 2$	2100
	$t = 3$	36	38		$t = 3$	1500	1300		$t = 3$	2000
$j = 3$	$t = 1$	50	48	$j = 3$	$t = 1$	2000	1600	$j = 3$	$t = 1$	3500
	$t = 2$	46	45		$t = 2$	1400	1850		$t = 2$	3200
	$t = 3$	43	43		$t = 3$	2100	2200		$t = 3$	3200

TABLE 6: Values of p_{ij} , t_{ij} , u_{ij} , and a_{ij} .

p_{ij}	$i = 1$	$i = 2$	t_{ij}	$i = 1$	$i = 2$	u_{ij}	$i = 1$	$i = 2$	a_{ij}	$i = 1$	$i = 2$
$j = 1$	0.003	0.005	$j = 1$	0.09	0.07	$j = 1$	21	20	$j = 1$	60	30
$j = 2$	0.006	0.004	$j = 2$	0.16	0.10	$j = 2$	24	25	$j = 2$	70	45
$j = 3$	0.007	0.009	$j = 3$	0.007	0.008	$j = 3$	28	27	$j = 3$	50	40

TABLE 7: Optimal values of x_{ijtr} and y_{ijtr} .

x_{ijtr}		$i = 1$	$i = 2$	y_{ijtr}		$i = 1$	$i = 2$
$j = 1$	$t = 1$	$r = 1$	—	$j = 1$	$t = 1$	$r = 1$	0
		$r = 2$	2000			$r = 2$	1
	$t = 2$	$r = 1$	—		$t = 2$	$r = 1$	0
$j = 1$		$r = 2$	600	$j = 1$		$r = 2$	1
	$t = 3$	$r = 1$	200		$t = 3$	$r = 1$	1
		$r = 2$	800			$r = 2$	0
$j = 2$	$t = 1$	$r = 1$	—	$j = 2$	$t = 1$	$r = 1$	0
		$r = 2$	500			$r = 2$	1
	$t = 2$	$r = 1$	—		$t = 2$	$r = 1$	0
$j = 2$		$r = 2$	500	$j = 2$		$r = 2$	1
	$t = 3$	$r = 1$	—		$t = 3$	$r = 1$	0
		$r = 2$	866			$r = 2$	1
$j = 3$	$t = 1$	$r = 1$	—	$j = 3$	$t = 1$	$r = 1$	0
		$r = 2$	1900			$r = 2$	1
	$t = 2$	$r = 1$	—		$t = 2$	$r = 1$	0
$j = 3$		$r = 2$	1350	$j = 3$		$r = 2$	1
	$t = 3$	$r = 1$	—		$t = 3$	$r = 1$	0
		$r = 2$	1700			$r = 2$	1

TABLE 8: Size of test instance sets.

Set no.	Number of OEMs	Number of products	Number of periods	Number of intervals
Set 1	3	4	3	2
Set 2	5	5	3	2
Set 3	8	10	4	3
Set 4	10	15	4	3
Set 5	20	20	5	4
Set 6	30	25	5	4
Set 7	40	30	5	5
Set 8	50	35	6	6
Set 9	80	40	6	6
Set 10	100	50	6	6

TABLE 9: Value range of parameters.

Parameter	Value range	Parameter	Value range
Production capacity	[500, 5000]	Original price	[10, 100]
Defective rate	[0.001, 0.01]	Demand	[1000, 10000]
Maximum acceptable defective rate	[0.005, 0.01]	Maximum acceptable late delivery rate	[0.05, 0.20]
Transaction cost	[30, 100]	Late delivery rate	[0.02, 0.20]
Unit carbon emission	[20, 30]	Discount rate	[0, 0.3]

$$\text{GAP} = \frac{F_{\text{GA}} - F_{\text{LINGO}}}{F_{\text{LINGO}}} \times 100\%. \quad (20)$$

Table 10 summarizes the computational results by solving each instance set by IGAUPE, GAUBE, GAUTE, and LINGO, and the average GAP and maximum GAP are also reported. We can draw some findings from Table 10.

For the first eight instance sets, the optimal solutions can be obtained by LINGO within 120 minutes. However, LINGO fails to find the optimal solutions of instance sets 9 and 10 in 120 minutes. Therefore, for the latter two instance sets, the current best feasible solutions are obtained by clicking the button of interrupt solver to terminate the LINGO program.

With the expansion of the instance set scale, the computation time of LINGO accordingly increases. For the latter five instance sets, the computation time of LINGO exceeds

10 minutes, which is the upper limit time for IGAUPE. However, all instance sets can be successfully solved by the proposed IGAUPE in the time limit of 10 minutes. Therefore, the proposed IGAUPE outperforms LINGO with respect to the computation time when solving large-scale instances.

All instance sets can be successfully solved by the proposed IGAUPE, GAUBE, and GAUTE. It is worth mentioning that the average GAP of proposed IGAUPE is on average 0.75% which is less than that of GAUBE and GAUTE.

TABLE 10: Comparison results of the IGAUPE, GAUBE, GAUTE, and LINGO.

Instance	Average GAP (%)			Maximum GAP (%)			LINGO average runtime (min)
	IGAUPE	GAUBE	GAUTE	IGAUPE	GAUBE	GAUTE	
Set 1	0.21	0.13	0.26	2.56	1.08	1.49	0.083
Set 2	0.39	0.67	0.75	2.14	2.90	3.17	0.217
Set 3	0.52	1.24	1.48	3.37	3.36	3.65	0.783
Set 4	0.78	3.86	1.97	4.68	6.50	6.02	2.25
Set 5	0.69	2.54	2.14	3.79	5.31	5.68	9.817
Set 6	0.85	3.11	2.39	3.72	6.29	5.84	19.367
Set 7	1.46	3.60	2.87	4.55	8.14	6.91	39.5
Set 8	1.93	4.29	3.01	4.63	7.25	6.33	82.533
Set 9	0.44	0.65	0.82	0.75	0.92	1.24	>120
Set 10	0.27	0.38	0.72	0.54	0.61	1.32	>120
Mean value	0.75	2.05	1.64	3.07	4.24	4.17	—

FIGURE 6: Average GAP for each instance set using IGAUPE, GAUBE, and GAUTE.

FIGURE 7: Maximum GAP for each instance set using IGAUPE, GAUBE, and GAUTE.

The maximum GAP of the proposed IGAUPE is on average 3.07% which is less than that of GAUBE and GAUTE. Specifically, the average GAP range of the proposed IGAUPE

is 0.21%~1.93%, the average GAP range of GAUBE is 0.13%~4.29%, and the average GAP range of GAUTE is 0.26%~3.01%. In terms of the maximum GAP index, the range of the proposed IGAUPE is 0.54%~4.48%, the range of GAUBE is 0.61%~8.14%, and the range of GAUTE is 1.24%~6.91%.

In brief, it is concluded that the proposed IGAUPE is significantly able to improve the performance and computation time compared with the GAUBE, GAUTE, and LINGO. Figures 6 and 7 demonstrate the average GAP and maximum GAP, respectively.

6. Conclusions

This paper focuses on modeling and solving the multi-objective sustainable order allocation problem. The main contributions of our work can be summarized as the following two aspects. First, considering product quality, carbon emission, quantity discounts, and multiperiod, the paper explores a multiproduct and multi-OEM integer nonlinear programming model. From the perspective of minimizing transaction cost, purchasing cost, number of defective products, number of late delivery product, and carbon emission of all orders, a novel multiobjective model is presented and converted to a single objective model using the weighting method. Second, due to the fact that the problem is NP-hard in nature, IGAUPE with a better encoding rule is proposed. Compared with GAUBE, GAUTE, and LINGO software, the performance evaluation of the proposed IGAUPE reveals that the proposed IGAUPE can improve the efficiency and effectiveness within the predetermined time limit when solving large-scale instances.

There are still some limitations in this work. The following directions can be extended in the future. First, it is not comprehensive to only consider the product defective rate in terms of quality. More quality factors should be considered in the future work. Second, in terms of sustainability, only carbon emission is considered in the paper. In the future, it is possible to take into account more sustainable criteria, such as energy-saving, green technology, recycled packaging, and utilization rate of resources. Nevertheless, our research can provide some references for future researches on modeling and solving the sustainable order allocation problem.

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 there are no conflicts of interest.

Acknowledgments

This research was funded by the National Natural Science Foundation of China (grant no. 71502050), Youth Foundation of Social Science and Humanity, China Ministry of Education (grant no. 19YJC630178), Philosophy and Social Science Foundation of Henan Province, China (grant no. 2018BJJ023), and Key Scientific and Technological Research Projects of Henan Provincial Department of Education, China (grant no. 15A630066).

References

- [1] N. Aissaoui, M. Haouari, and E. Hassini, "Supplier selection and order lot sizing modeling: a review," *Computers & Operations Research*, vol. 34, no. 12, pp. 3516–3540, 2007.
- [2] A. Mendoza and J. A. Ventura, "An effective method to supplier selection and order quantity allocation," *International Journal of Business and Systems Research*, vol. 2, no. 1, pp. 1–15, 2008.
- [3] J. Bull, "Loads of green washing-can behavioural economics increase willingness-to-pay for efficient washing machines in the UK?," *Energy Policy*, vol. 50, pp. 242–252, 2012.
- [4] C. Dong, B. Shen, P.-S. Chow, L. Yang, and C. T. Ng, "Sustainability investment under cap-and-trade regulation," *Annals of Operations Research*, vol. 240, no. 2, pp. 509–531, 2016.
- [5] Z. X. Guo, W. K. Wong, S. Y. S. Leung, J. T. Fan, and S. F. Chan, "Genetic optimization of order scheduling with multiple uncertainties," *Expert Systems with Applications*, vol. 35, no. 4, pp. 1788–1801, 2008.
- [6] Z. X. Guo, W. K. Wong, and S. Y. S. Leung, "A hybrid intelligent model for order allocation planning in make-to-order manufacturing," *Applied Soft Computing*, vol. 13, no. 3, pp. 1376–1390, 2013.
- [7] M. Seifbarghy and N. Esfandiari, "Modeling and solving a multi-objective supplier quota allocation problem considering transaction costs," *Journal of Intelligent Manufacturing*, vol. 24, no. 1, pp. 201–209, 2013.
- [8] S. Nazari-Shirkouhi, H. Shakouri, B. Javadi, and A. Keramati, "Supplier selection and order allocation problem using a two-phase fuzzy multi-objective linear programming," *Applied Mathematical Modelling*, vol. 37, no. 22, pp. 9308–9323, 2013.
- [9] P. Renna and G. Perrone, "Order allocation in a multiple suppliers-manufacturers environment within a dynamic cluster," *The International Journal of Advanced Manufacturing Technology*, vol. 80, no. 1–4, pp. 171–182, 2015.
- [10] S. Gupta, I. Ali, and A. Ahmed, "Multi-objective bi-level supply chain network order allocation problem under fuzziness," *OPSEARCH*, vol. 55, no. 3–4, pp. 721–748, 2018.
- [11] V. Jain, A. Kundu, F. T. S. Chan, and M. Patel, "A chaotic bee colony approach for supplier selection-order allocation with different discounting policies in a cooperative multi-echelon supply chain," *Journal of Intelligent Manufacturing*, vol. 26, no. 6, pp. 1131–1144, 2015.
- [12] L. Duan and J. A. Ventura, "A dynamic supplier selection and inventory management model for a serial supply chain with a novel supplier price break scheme and flexible time periods," *European Journal of Operational Research*, vol. 272, no. 3, pp. 979–998, 2019.
- [13] M. Alegoz and H. Yapicioglu, "Supplier selection and order allocation decisions under quantity discount and fast service options," *Sustainable Production and Consumption*, vol. 18, pp. 179–189, 2019.
- [14] S. Hosseini, N. Morshedlou, D. Ivanov, M. D. Sarder, K. Barker, and A. A. Khaled, "Resilient supplier selection and optimal order allocation under disruption risks," *International Journal of Production Economics*, vol. 213, pp. 124–137, 2019.
- [15] M. Goh and S. PrasannaVenkatesan, "Multi-objective supplier selection and order allocation under disruption risk," *Transportation Research Part E*, vol. 95, pp. 124–142, 2016.
- [16] S. Li, "Genetic algorithm to logistics service supply chain order allocation problem," *Operations Research and Management Science*, vol. 23, pp. 35–41, 2014, in Chinese.
- [17] X. Hu, G. Wang, X. Li, Y. Zhang, S. Feng, and A. Yang, "Joint decision model of supplier selection and order allocation for the mass customization of logistics services," *Transportation Research Part E: Logistics and Transportation Review*, vol. 120, pp. 76–95, 2018.
- [18] P. Gupta, K. Govindan, M. K. Mehlaawat, and S. Kumar, "A weighted possibilistic programming approach for sustainable vendor selection and order allocation in fuzzy environment," *The International Journal of Advanced Manufacturing Technology*, vol. 86, no. 5–8, pp. 1785–1804, 2016.
- [19] K. Govindan and R. Sivakumar, "Green supplier selection and order allocation in a low-carbon paper industry: integrated multi-criteria heterogeneous decision-making and multi-objective linear programming approaches," *Annals of Operations Research*, vol. 238, no. 1–2, pp. 243–276, 2016.
- [20] M. K. Ghorabae, M. Amiri, E. K. Zavadskas, Z. Turskis, and J. Antucheviciene, "A new multi-criteria model based on interval type-2 fuzzy sets and EDAS method for supplier evaluation and order allocation with environmental considerations," *Computer & Industrial Engineering*, vol. 112, pp. 156–174, 2017.
- [21] C. Babbar and S. H. Amin, "A multi-objective mathematical model integrating environmental concerns for supplier selection and order allocation based on fuzzy QFD in beverages industry," *Expert Systems with Applications*, vol. 92, pp. 27–38, 2018.
- [22] S. Hamdan and A. Cheaitou, "Supplier selection and order allocation with green criteria: an MCDM and multi-objective optimization approach," *Computers & Operations Research*, vol. 81, pp. 282–304, 2017.
- [23] A. Cheaitou and S. Hamdan, "Dynamic green supplier selection and order allocation with quantity discounts and varying supplier availability," *Computer & Industrial Engineering*, vol. 110, pp. 573–589, 2017.
- [24] P. Ghadimi, F. G. Toosi, and C. Heavey, "A multi-agent systems approach for sustainable supplier selection and order allocation in a partnership supply chain," *European Journal of Operational Research*, vol. 269, no. 1, pp. 286–301, 2018.
- [25] H. Moheb-Alizadeh and R. Handfield, "Sustainable supplier selection and order allocation: a novel multi-objective programming model with a hybrid solution approach," *Computers & Industrial Engineering*, vol. 129, pp. 192–209, 2019.

- [26] A. Mohammed, I. Harris, and K. Govindan, "A hybrid MCDM-FMOO approach for sustainable supplier selection and order allocation," *International Journal of Production Economics*, vol. 217, pp. 171–184, 2019.
- [27] W. Liu, D. Wang, X. Shen, X. Yan, and W. Wei, "The impacts of distributional and peer-induced fairness concerns on the decision-making of order allocation in logistics service supply chain," *Transportation Research Part E: Logistics and Transportation Review*, vol. 116, pp. 102–122, 2018.
- [28] Z. Gao, W. Liu, Z. Fan, and W. Zhu, "Multi-objective optimization model of order allocation of LSSC considering transaction costs," *Systems Engineering*, vol. 30, pp. 35–39, 2012, in Chinese.
- [29] S. M. Arabzad, M. Ghorbani, J. Razmi, and H. Shirouyehzad, "Employing fuzzy TOPSIS and SWOT for supplier selection and order allocation problem," *The International Journal of Advanced Manufacturing Technology*, vol. 76, no. 5–8, pp. 803–818, 2015.

Submit your manuscripts at
www.hindawi.com