Iranian Journal of Management Studies (IJMS) *http://ijms.ut.ac.ir/*
Vol. 12, No. 2, Spring2019 *Print ISSN: 2008-7055*
pp. 281-306 *Online ISSN: 2345-3745*
DOI: 10.22059/IJMS. 2019.254633.673069

A Genetic Algorithm Developed for a Supply Chain Scheduling Problem

Seyed Mohammad Reza Taheri, Mohammad Ali Beheshtinia*

Department of Industrial Engineering, University of Semnan, Semnan, Iran

(Received: May 22, 2018; Revised: December 10, 2018; Accepted: December 17, 2018)

Abstract
This paper concentrates on the minimization of total tardiness and earliness of orders in an integrated production and transportation scheduling problem in a two-stage supply chain. Moreover, several constraints are also considered, including time windows due dates, and suppliers and vehicles availability times. After presenting the mathematical model of the problem, a developed version of GA called Time Travel to History (TTH) algorithm, inspired from the idea of traveling through history, is proposed to solve the problem. In order to validate the performance of the proposed algorithm, the results of TTH algorithm are compared with two other genetic algorithms in the literature. The comparison results show the better performance of the proposed algorithm. Moreover, the results of implementing the sensitivity analysis to the main parameters of the algorithm show the behavior of the objective functions when the parameters are changed.

Keywords
Genetic algorithm, Meta-heuristic, Supply chain, Scheduling, Logistic.

***Corresponding Author, Email:** beheshtinia@semnan.ac.ir

Introduction

As the strong competition in different industries is growing rapidly, firms are compelled to cooperate with their supply chain members in order to improve the competitiveness of their final product. A supply chain is a set of suppliers, manufacturers and distributors, pursuing the same ultimate objective, that is, meeting the final customers' needs, which results in creating added value to the products (Y.-C. Chang & Lee, 2004). Making manufacturing and transportation decisions separately and unrelated to one another does not lead to an optimal solution. The reason is that, the decisions made in the manufacturing and transportation section are interrelated and may influence both sections, simultaneously. Integration in manufacturing and transportation decisions brings about increased cooperation, greater efficiency in the supply chain, reduced manufacturing time and costs, lower quality control costs, improvements in delivery times, enhancement of product quality, and eventually, increased final customer satisfaction (Kumar, Vrat, & Shankar, 2004).

This paper is focused on the manufacturing and transportation scheduling problem in a supply chain, comprising one manufacturer and multiple suppliers. The manufacturer assigns its orders to the suppliers to be processed and the processed orders are transported to the manufacturer via a shared transportation fleet through a Vehicle Routing Problem (VRP) approach. In VRP, different orders, which may belong to different suppliers, are allowed to be carried by one vehicle in one batch (route) and then, be delivered to the manufacturer. This approach reduces transportation costs and results in more efficient usage of the vehicles. In addition, several constraints are considered in this problem, including delivery time windows and availability times for the suppliers and vehicles. Timely delivery of the products to the manufacturer plays a key role in the manufacturing of the final products. Delivery of orders later than their due dates may cause manufacturing disruptions and even interruptions in the production line. Also, when the orders are delivered sooner than their due date, the excessive storage of goods incurs extra inventory costs. This assumes even greater significance in perishable products supply chains, such as food, medicine and chemicals. Therefore, the objective function of the problem in this study is to simultaneously minimize the total tardiness and earliness of orders delivery to the manufacturer.

Once a mathematical model for the problem is proposed, an extension of GA, called Time Travel to History (TTH) algorithm, is proposed to solve it. This algorithm is inspired from the concept of traveling between different positions in time. In the proposed algorithm, particular chromosomes are transported to the previous generations, if some predetermined conditions are met.

The main aim of this study is minimizing the total tardiness and earliness of orders in an integrated production and transhipment scheduling problem in a supply chain. The main contribution of this study is proposing a new version of GA namely Time Travel to History (TTH) algorithm, inspired from the idea of traveling through history to solve the problem.

In the remainder of this paper, a literature review on this field is presented in section 2. The problem specifications, mathematical model of the problem and research steps are presented in section 3. In section 4, a new algorithm is proposed to solve it. The proposed algorithm is evaluated and validated in section 5 and eventually, concluding discussions and suggestions for future research are made in the last section.

Literature review

Numerous researches have focused on supply chain scheduling.

Zegordi and Beheshti Nia (2009) intended to integrate manufacturing and transhipment scheduling in a supply chain to assign orders to manufacturers. A mathematical model was proposed for the problem and GA was used to solve it.

Yeung et al. (2011) focused on minimizing the inventory and transportation costs in a supply chain scheduling problem. They used multiple time windows for products delivery. Fahimnia et al. (2012) used non-linear integer mathematical programming for integrated manufacturing-distribution in a 2-stage supply chain with real-world variables and constraints. Ullrich (2013) considered time windows in their study about the integration of machines scheduling and vehicles routing in a 2-stage supply chain. The first stage included a parallel-machine environment, where setup times depended on the machines, and the second stage was comprised of a transportation fleet, where vehicles had different capacities. Selvarajah and Zhang (2014) aimed to schedule a supply chain in which the semi-finished materials were collected by the manufacturer from the suppliers at different times. Then, the manufacturer delivered the finished products to the customers in batches.

Han and Zhang(2015) studied on-line scheduling in supply chain with single machine and multiple customers. They also considered several constraints such as unlimited number of vehicles and limited vehicle capacity. They aimed to minimize the total makespan and the total delivery cost.

The integrated manufacturing and distribution problem was studied by Chang et al. (2016), considering orders to be processed by unrelated parallel machines without being stored in the production stage and then, delivered to the customers by vehicles with limited capacity. The goal was to reduce the total cost, considering customer service level and the total distribution cost. Yin et al. (2016) presented a bi-objective mathematical formula for the cross-dock problem, in which vehicles are planned to achieve the highest throughput. Their problem included multiple vehicles with limited capacities which deliver the orders to the receiving door of cross-dock. Once the orders are collected and placed into the cross-dock by forklifts or conveyors, they are loaded into order vehicles in the shipping door of cross-dock and delivered to the final customer.

Karaoğlan and Kesen (2017) intended to integrate the production and transportation decisions in short lifespan production. The products were distributed to the customers by a single vehicle having limited capacity before the lifespan. The objective function was to determine the minimum time required to produce and deliver all customer demands. They designed a branch-and-cut algorithm for the problem. The shared transportation problem and multi-site manufacturing scheduling problem were both considered by Beheshtinia et al. (2017). They proposed a GA, inspired by the role model concept in sociology to solve the problem. They presented a mathematical model for the problem and implemented the proposed algorithm on a pharmaceutical company in Iran. Beheshtinia and Ghasemi (2017) also presented an integrated multi-objective model to optimize supply chain scheduling in a multi-site manufacturing system. They focused on a problem, comprised of multiple suppliers and vehicles. The vehicles deliver raw materials from the suppliers to the manufacturers. The objective function of the problem was to simultaneously minimize the tardiness of the orders assigned to the suppliers, and the distance travelled by the vehicles. They used the multiple league championship algorithm to solve the problem.

Xu et al. (2017) studied the feature of the transportation scheduling problem in a supply chain with a third-party logistics enterprise. They categorized all transportation nodes into three groups, and presented an Ant Colony optimization algorithm with negative selection operation for each, according to the type of the transportation in the nodes.

Borumand and Beheshtinia (2018) proposed a new algorithm, based on the mixture of GA and VIKOR, for solving an integrated manufacturing and transportation scheduling problem in a supply chain, with multiple objective functions.

These studies may be grouped in terms of different aspects. In terms of integration between elements of supply chain, these studies are categorized into four groups: 1) the researches that explore transactions between manufacturers and their suppliers; 2) the researches that examine transactions between the manufacturers and their distributors or their customers; 3) transactions between several manufacturers, and 4) combination of the above scenarios. In terms of considering transportation, the studies are categorized into two groups: 1) the studied that considered transportation, and 2) the studies that did not consider transportation. However, there are also papers which despite mentioning the word *transportation* in their titles and texts, consider it only as a fixed time or cost. Hence, these studies have not practically

considered transportation constraints. Table 1 represents the categorization of these studies.

Table 1.Categorized researches of the literature

Article	**Integration Level**				**Defined transportation navigation**		**Objective function**		**Availability time**
	Suppliers - manufacturer	Manufacturer-distributor	Multiple parallel manufacturers	Hybrid	Yes	No	Earliness	Tardiness	
Moon et al.(2008)			*		*				
Zegordi and Beheshti Nia(2009)	*				*				
Yimer and Demirli (2010)				*		*			
Yeung et al.(2011)				*		*		*	
Fahimnia et al. (2012)		*			*				
Ullrich (2013)	*				*			*	*
Selvarajah and Zhang (2014)		*				*			
Han and Zhang(2015)		*			*				
Chang et al. (2016)		*			*				
Yin et al. (2016)				*	*				
Karaoğlan and Kesen (2017)		*			*				
Low et al.(2017)		*			*				
Beheshtinia et al. (2017)	*				*				
Beheshtinia and Ghasemi (2017)	*				*			*	*
Xu et al. (2017)				*	*				
Borumand and Beheshtinia (2018)	*				*			*	
our research	*				*		*	*	*

The literature review shows that no research has been done regarding the integration of manufacturing scheduling in suppliers and transportation scheduling in a supply chain with the objective function of simultaneously minimizing the total tardiness and total earliness. The availability times for suppliers and vehicles are also considered as a constraint in this study. Moreover, a developed genetic algorithm, named TTH algorithm, is used to solve the problem. Therefore, the innovations of this paper are as follows:

- Considering the integration of suppliers manufacturing and transportation scheduling in a supply chain, in order to simultaneously minimize the total tardiness and earliness.
- Presenting a mathematical model for the problem.
- Employing an extension of GA, inspired from the concept of time travel between different points of time, in order to solve the problem.

In the next section, the problem and its features are described in detail and the mathematical model is presented.

Problem specifications and research steps

In this section, first the problem attributes are described. Then, the mathematical model of the problem is proposed.

Problem features

In this paper, the simultaneous manufacturing and transhipment scheduling is studied in a supply chain. The supply chain is comprised of a manufacturer and some suppliers. In what follows, the problem is described in further details:

- There are n orders which should be assigned to m suppliers to be processed. Once they are processed by their corresponding suppliers, they should be collected and transported to the manufacturer, using v vehicles.
- Each order should be assigned to and completed by precisely one supplier.
- Each order has a process time, a certain weight, and a delivery time window. The delivery time window is an interval, indicated by (a,b). If the order is delivered later than the upper bound (b), then tardiness has occurred. Similarly, earliness is caused when the order is delivered sooner than the lower bound (a).
- The transportation fleet is heterogeneous, that is, the vehicles have different and limited speeds and capacities. The average speeds of the vehicles remain constant during the scheduling period and are different from each other.
- At the start of the scheduling, all vehicles are located at the same terminal.
- The distances between the suppliers as well as their distance to the terminal and the manufacturer are predetermined.
- Similar to VRP, in order to optimize the usage of the transportation fleet, it is shared among the suppliers; and each vehicle is allowed to carry the orders, assigned to different suppliers, in a single route (batch) and deliver them to the manufacturer. Moreover, once the vehicles deliver the batch to the manufacturer, they are not eliminated from the problem and may be used again.

- The loading times of the orders may be different for each batch. The loading time of an order equals the maximum value of the two amounts of the order's completion time and the vehicle's arrival time. However, the orders delivery times to the manufacturer are equal in a batch.
- An availability time is considered for each supplier and each vehicle. Vehicles and suppliers are available after the availability time. This allows for rescheduling, when a disruption happens in the current scheduling. In case a disruption occurs, some suppliers or vehicles may be unavailable for a specific amount of time. For example, due to the machinery failure or the production line being busy with processing previously-assigned orders (the frozen zone), a supplier might be unavailable even at the start of the scheduling. A similar situation may also happen to the vehicles. Therefore, an availability time should be considered for the vehicles and the suppliers.

The goal is to decide how orders are assigned to suppliers and vehicles, and to determine the processing sequence of the orders that are assigned to the suppliers, in a way that the total tardiness and total earliness are minimized.

For further clarification, an example is illustrated in Figure 1, in which two vehicles are responsible for collecting the five processed orders from four suppliers. First, the first vehicle collects order 4 from supplier 4. Then, as it still has some unused capacity, it collects order 2 from supplier 1 and returns to the manufacturer for delivery. The other vehicle first collects orders 1 and 5 from supplier 3, and then fills its remaining empty capacity by collecting order 3 from supplier 2 and returns to the manufacturer.

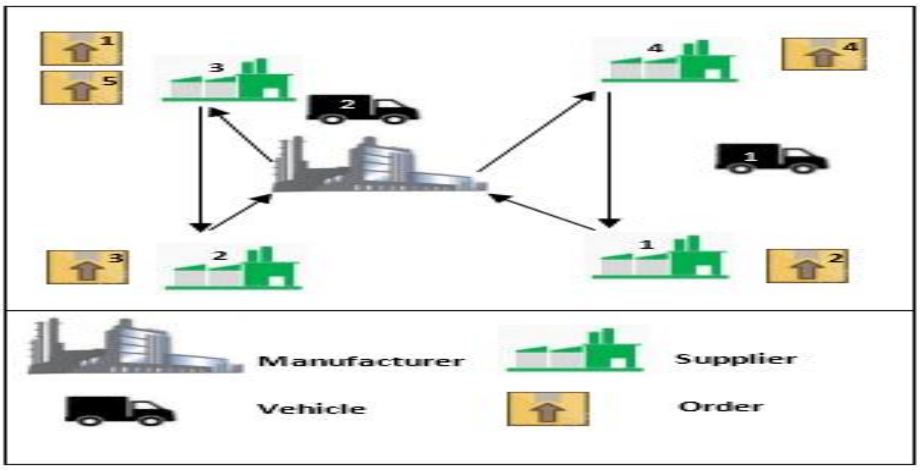


Figure 1. a feasible solution for the problem.

The mathematical model

In this section, the mathematical model of the problem is presented. The parameter notations and decision variables of the problem are as follows:

Ns	Number of suppliers
No	Number of orders
Nv	Number of vehicles
q or i	Order index
s or s'	Supplier index
b	Batch index
p	Transportation priority index for the orders of a batch
k	Vehicle index
pt_i	Process time of order *i*
$SupInAv_s$	Initial availability of supplier *s*
$VehInAv_k$	Initial availability of vehicle *k*
Siz_i	Occupied space by order *i* on a vehicle
VS_k	Velocity of vehicle *k*
Cap_k	Capacity of vehicle *k*
DIS_s	Distance between supplier *s* and the manufacturer
$DIST_s$	Distance from the terminal to supplier *s*
$DISS_{ss'}$	Distance between supplier s to supplier s'
$Udue_i$	Upper bound of the due date time window
$Ldue_i$	Lower bound of the due date time window
pt_i	Process time of order *i*
M	A large positive number

The variables are introduced as follows:

$Delivery_i$	Delivery time of order *i*
$Tardiness_i$	Tardiness of order *i*
$Earliness_i$	Earliness of order *i*
co_i	Completion time of order *i* by suppliers
$Load_i$	Loading time of order *i* by the related vehicle
V_{kbip}	Equals 1, if order *i* has the p^{th} transfer priority in the b^{th} batch of vehicle *k*; otherwise, it equals 0
x_{is}	Equals 1, if order *i* is assigned to supplier *s*; otherwise, it equals 0
Av_{kbi}	Availability of the vehicle *k* to transfer order *i* in batch *b*
y_{iq}	Equals 1, if order *i* has higher production priority than order *q* at the supplier stage; otherwise, it equals 0

The mathematical model of the problem is presented as follows:

$$MinZ = \sum_{i=1}^{No} Tardiness_i + \sum_{i=1}^{No} Earliness_i \tag{1}$$

S.t.:

$$\sum_{s=1}^{Ns} x_{si} = 1 \qquad i = 1; 2; \dots; No \tag{2}$$

$$\sum_{k=1}^{Nv} \sum_{b=1}^{No} \sum_{p=1}^{No} V_{kbip} = 1 \qquad i = 1; 2; \dots; No \tag{3}$$

$$\sum_{i=1}^{No} V_{kbip} \leq 1 \qquad k = 1; 2; \dots; Nv \quad b = 1; 2; \dots; No \quad p = 1; 2; \dots; No \tag{4}$$

$$\sum_{i=1}^{No} \sum_{p=1}^{No} SIZ_i \times V_{kbip} \leq Cap_k \qquad k = 1; 2; \dots; Nv \quad b = 1; 2; \dots; No \tag{5}$$

$$co_i \geq SupInAv_s + Pt_i - M(1 - x_{is}) \qquad i = 1; 2; \dots; No \quad s = 1; 2; \dots; Ns \tag{6}$$

$$\begin{aligned} co_i + M * \left(2 + y_{iq} - x_{si} - x_{sq}\right) &\geq co_q + Pt_i \\ co_q + M * \left(3 - y_{iq} - x_{si} - x_{sq}\right) &\geq co_i + Pt_q \end{aligned} \qquad i; q = 1; 2; \dots; No \quad i < q \quad s = 1; 2; \dots; Ns \tag{7}$$

$$y_{iq} = 0 \qquad i; q = 1; 2; \dots; No \quad i \geq q \tag{8}$$

$$\sum_{i=1}^{No} V_{kbi(p+1)} \leq \sum_{i=1}^{No} V_{kbip} \qquad k = 1; 2; \dots; Nv \quad b = 1; 2; \dots; No \quad p = 1; 2; \dots; No - 1 \tag{9}$$

$$\sum_{i=1}^{No} V_{k(b+1)i1} \le \sum_{i=1}^{No} V_{kbi1} \qquad k = 1;2;\dots;Nv \quad b = 1;2;\dots;No-1 \tag{10}$$

$$Load_i \ge Av_{kbi} - M(1 - \sum_{p=1}^{No} V_{kbip}) \qquad i = 1;2;\dots;No \quad k = 1;2;\dots;Nv \quad b = 1;2;\dots;No \tag{11}$$

$$Load_i \ge co_i \qquad i = 1;2;\dots;No \tag{12}$$

$$av_{k1i} \ge VehInAv_k + \frac{DIST_S}{VS_k} - M(2 - V_{k1i1} - x_{is}) \qquad i = 1;2;\dots;No \quad k = 1;2;\dots;Nv \quad s = 1;\dots;Ns \tag{13}$$

$$av_{kbi} \ge delivery_q + \frac{DIS_S}{VS_k} - M * (3 - V_{kbi1} - V_{k(b-1)q1} - x_{is}) \qquad i;q = 1;2;\dots;No \quad k = 1;2;\dots;Nv \quad s = 1;\dots;Ns \quad b = 2;\dots;No \tag{14}$$

$$av_{kbi} \ge Load_q + \frac{DISS_{s's}}{VS_k} - M * (4 - V_{kbqp} - V_{kbi(p+1)} - x_{qs'} - x_{is}) \qquad i;q = 1;2;\dots;No \quad k = 1;2;\dots;Nv \quad s;s' = 1;\dots;Ns \quad b = 1;2;\dots;No \quad p = 1;2;\dots;No-1 \tag{15}$$

$$delivery_i \ge Load_q + \frac{DIS_S}{VS_k} - M * (3 - \sum_{p=1}^{No} V_{kbip} - \sum_{p=1}^{No} V_{kbqp} - x_{qs}) \qquad k = 1;2;\dots;Nv \quad b = 1;2;\dots;No \quad i;q = 1;2;\dots;No \quad s = 1;\dots;Ns \tag{16}$$

$$Tardiness_i \ge delivery_i - Udue_i \qquad i = 1;2;\dots;No \tag{17}$$

$$Earliness_i \ge Ldue_i - delivery_i \qquad i = 1;2;\dots;No \tag{18}$$

$$\begin{array}{llll} Earliness_i \ge 0 & \forall i & co_i \ge 0 & \forall i \\ V_{kbip} \ge 0 & \forall k;b;i;p & Delivery_i \ge 0 & \forall i \\ x_{si} \in \{0;1\} & \forall i;s & Load_i \ge 0 & \forall i \\ y_{iq} \in \{0;1\} & \forall i;q & Av_{kbi} \ge 0 & \forall k;b;i \\ & & Tardiness_{i \ge 0} & \forall i \end{array} \tag{19}$$

This model is a single-objective mathematical model, and equation (1) represents the objective function, which consists of two parts: 1) minimizing the total tardiness of orders and 2) minimizing the total earliness of orders. Constraint (2) ensures that each order must be allocated to exactly one supplier. Constraint (3) ensures that each order must be allocated to one priority of one batch of one vehicle. Constraint (4) states that two orders must not be allocated to one priority of a batch. Constraint (5) states that the total occupied space by the allocated orders to each batch of a vehicle should be lower than the vehicle's capacity. Constraint (6) considers the relationship between the completion time and processing time of an order. Constraint set (7) restricts each supplier to processing only one order at a time. Some extra variables are removed by Constraint (8). Constraint (9) specifies that if no order is allocated to priority p of batch b, then it is not possible to allocate an order to priority p+1 of the batch. Constraint (10) indicates that if there is no assignment to batch b, then it is not possible to allocate an order to batch b+1. Constraint (11) links the loading time of an order to the availability time of the corresponding vehicle. Constraint (12) links the loading time of an order to its completion time. Constraint (13) controls a vehicle's availability time to transport the first order of its first batch. Constraint (14) determines a vehicle's availability time to transport the first order of its other batches. Constraint (15) describes the link between the vehicle availability time of an order and the previously-allocated orders' loading time. Constraint (16) ensures that the delivery time of the allocated orders to a batch are equal. Constraint (17) indicates the relationship between the tardiness, delivery time and due date of each order. Constraint (18) determines the earliness of each order.

The following steps are performed in this research:

Step 1: Employing the TTH algorithm to solve the problem

Step 2: Evaluating the performance of the proposed algorithm by comparing its results with two other algorithms

Step 3: Comparison of TTH with optimum solutions

Step 4: Performing a sensitivity analysis on the parameters of the algorithm

TTH algorithm

GA is a broadly-used algorithm to solve NP-hard problems, which was first introduced by John Holland (1992). First, a generation of random chromosomes (random solutions) are created to establish the initial population. Then, the two mutation and crossover operators are used to increase the population of the current generation. Afterwards, using the selection operator, a number of chromosomes are selected to proceed to the next generation. This procedure is repeated until the termination criterion of the algorithm is met.

In this study, a developed genetic algorithm, named TTH algorithm, is used to solve the problem. TTH algorithm is inspired by the concept of travelling back in time. Although time travel has not yet been possible in the real world, its concept is applicable to GA. For this purpose, some modifications are made to the conventional GA. When a certain criterion, called the travel criterion, is satisfied, a number of chromosomes are selected from a generation to be transferred to a few generations before. This procedure is repeated until the termination criterion of the algorithm is met. The main parameters of the proposed GA are as follows:

Pop_size: it indicates the initial population size.

Cross_rate: the repetition number of the crossover operation (Cross_rate* Pop_size) is defined by this parameter.

Mut-rate: the repetition number of the mutation operation (Mut_rate* Pop_size) is defined by this parameter.

STOP: it is an iteration number that indicates the termination criterion of the algorithm. If the best chromosome in the current generation is not improved by STOP successive iterations, then the algorithm is terminated.

ELIT: indicates the selection operator in this algorithm. A percentage of the chromosomes in the current generation with better objective function value than others are selected to proceed to the next generation. This percentage is indicated by ELIT. In other words, ELIT* Pop_size chromosomes with the most suitable objective function values are selected to directly proceed to the next generation. The remaining chromosomes (Pop_size -ELIT* Pop_size) are randomly selected to proceed to the next generation, using the roulette wheel selection.

In TTH algorithm, one more criterion and two more parameters are added to GA. They are described in the following lines:

Travel criterion: once the criterion is satisfied, the traveling process is initiated. In this study, the travel criterion is the generation counter (CURRENT). The criterion is met, when the generation counter reaches a certain number, indicated by GB.

Trans_rate: it determines the number of chromosomes in the current generation that should travel to previous generations. In other words, as many as Pop_size*Trans_rate of the best chromosomes of the current generations are selected and transferred to the previous generations.

R: This parameter determines how many generations the chromosomes should travel back. Let CURRENT be the number of the current generation. Then, the selected chromosomes should travel back to generation CURRENT-R. In this case, as many as Pop_size*Trans_rate of the worst chromosomes of generation CURRENT-R are eliminated and replaced by the selected chromosomes.

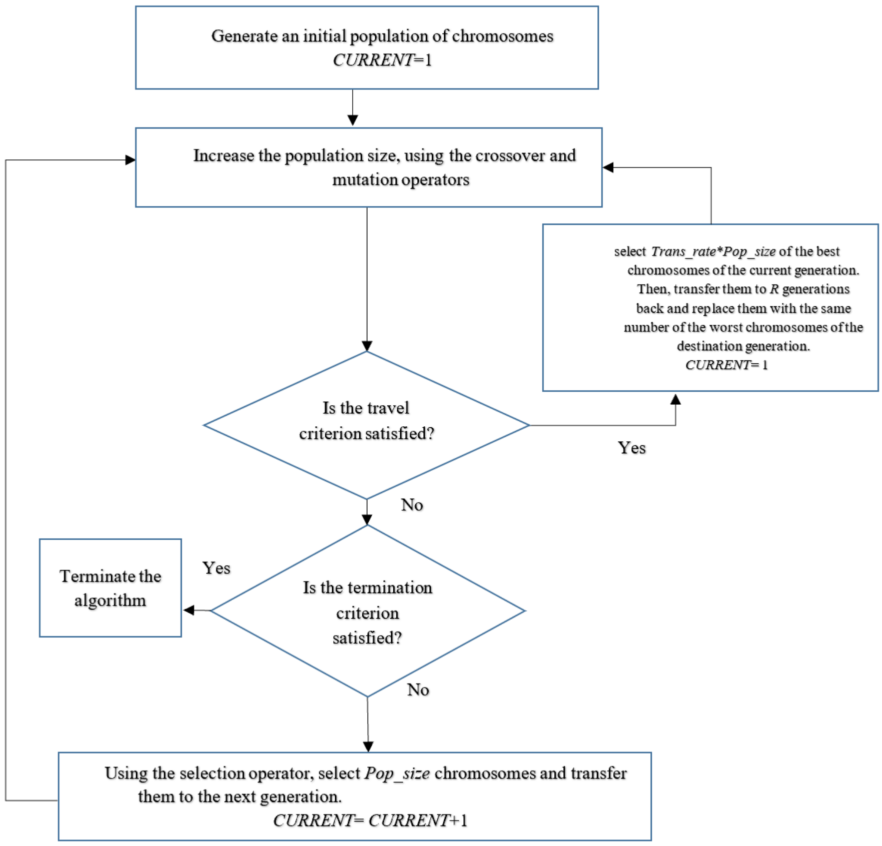


Figure 2. The flowchart of TTH algorithm.

The implementation steps of TTH algorithm are as follows:

Step 1- create a number of random chromosomes as the initial population and let CURRENT=1. (CURRENT is the number of the current generation).

Step 2- increase the population of the current generation, using mutation and crossover operators.

Step 3- if the travel criterion is met (CURRENT>GB), take step 6; otherwise, take step 4.

Step 4- if the termination criterion is met, terminate the algorithm; otherwise, take step 5.

Step 5- using the selection operator, select Pop_size chromosomes and transfer them to the next generation. Let CURRENT=CURRENT+1 and return to step 2.

Step 6 – select Trans_rate*Pop_size of the best chromosomes of the current generation and transfer them to R generations back. Then, replace them with the same number of the worst chromosomes of the destination generation. Let CURRENT=1 and return to step 2.

The crossover and mutation operators: the crossover and mutation operators, used in this study, are similar to those, used in the study of Ullirch(2013).

Using the Taguchi testing method, the following values are obtained for the parameters of the algorithm: Pop_size= 100, Cross_rate= 0.5, STOP= 15, Mut_rate= 0.5, ELIT= 0.01, GB= 10, R= 5, and Trans_rate= 0.2.

Results

Evaluating the performance of TTH algorithm

In order to evaluate the performance of the proposed algorithm, the results of the TTH algorithm are compared with the results of the GA, presented by Ullrich (2013), called $GA_{Ullrich}$, and another developed GA, which has the same structure as TTH algorithm, except for the time travel mechanism that is not used in it (Zegordi & Beheshti Nia, 2009), called OGA. In other words, OGA is a version of TTH algorithm, in which GB equals a large positive number, so that the travel criterion is never satisfied. Eventually, a sensitivity analysis is performed on three key parameters of the algorithm. In order to compare TTH and OGA, 27 problems are generated with different sizes. Due to the random nature of GA, it is likely to yield a dissimilar result in each run. Therefore, each algorithm is run for 20 times for each problem and its performance is evaluated using hypothesis testing.

$GA_{Ullrich}$ algorithm

The steps of $GA_{Ullrich}$ are as follows:

Step 1: Create an initial population randomly.

Step 2: Define the three operators: mutation, crossover and selection.

Step 3: Create the next generation by performing the following steps.

Step 3-1: Select one of the mutation, crossover or selection operators randomly with equal probability.

Step 3-2: Perform the selected operator and convey the result to the next generation.

Step 3-3: If the number of chromosomes in the next generation reached the initial population size, go to step 4; otherwise, go to Step 3-1.

Step 4: if the termination criterion is met, terminate the algorithm; otherwise, take step 3.

In OGA, the number of chromosomes in each generation increases from the initial population size by crossover and mutations operators, and then a number of chromosomes are selected among them to go to the next generation. But in $GA_{Ullrich}$, unlike popular algorithms, population size is constant in each generation, and only the chromosomes are transmitted in three ways to the next generation. The first method is the selection operator in which a chromosome is selected by the roulette wheel operator and transmitted directly to the next generation. In the second method, which is crossover, two chromosomes of the current generation are selected and applied to the merger, and the offspring are going to the next generation. In the third method, which is called mutation, a chromosome is selected, and after mutation, the mutated chromosome is transmitted to the next generation.

Generating test problems

The problem has ten major parameters: 1) number of orders, 2) number of suppliers, 3) number of vehicles, 4) process time of orders, 5) due date windows, 6) availability times of suppliers and vehicles, 7) velocity of vehicles, 8) transportation distances, 9) size of orders, and 10) capacity of vehicles. Three levels of high, medium and low are considered for the three main problem parameters, which are the total number of orders, suppliers and vehicles. The values of other parameters are determined using uniform distributions with specified ranges. Different levels of these parameters are given in Table 2.

Table 2. The values determined for the parameters of the test problems

	Low	Medium	High
Number of orders	10	50	100
Number of suppliers	1	10	20
Number of vehicles	1	10	20
Process time of orders	U[1,20]		
Transportation distances	U[1,20]		
Availability times of suppliers and vehicles	U[1,5]		
Size of orders	U[1,5]		
Velocity of vehicles	U[1,2]		
Capacity of vehicles	U[5,20]		
Due date windows	Lower bound= U[25,30] Upper bound= U[35,40]		

Considering all possible conditions (3*3*3*1*1*1*1*1*1*1), 27 random problems are generated, which are used as the test problems.

Comparison results

All 27 random problems are solved by the three algorithms and the results are compared with each other, using hypothesis testing. In this study, the coding is done using MATLAB, and it is run by a computer with an Intel Corei5 2.5GHz CPU. As mentioned before, each of these algorithms is run 20 times and the obtained results are compared by a hypothesis test with confidence level of 95% (1-α). For each of these 27 problems, two hypothesis tests are performed (a total of 54 tests). The hypotheses are as follows:

H_0: $\mu_{TTH} = \mu_{compared\ algorithm}$
H_1: $\mu_{TTH} < \mu_{compared\ algorithm}$

When H_0 is rejected, it means that the performance of TTH algorithm is better than the performance of the compared algorithm. The testing results are shown in Table 3. This table has three indexes:

- The average mean solutions, obtained by the algorithms
- The mean solving time of the algorithms (presented in seconds)
- The P-value of the tests.
- Improvement Rate

Table 3. Comparison of TTH with GAUllrich and OGA

Problem	Number of orders	Number of vehicles	Number of suppliers	TTH		OGA		$GA_{Ullrich}$		P-Value (TTH-OGA)	P-Value (TTH-$GA_{Ullrich}$)	Improvement Rate
				Avrg	Mean CPU Time	Avrg	Mean CPU Time	Avrg	Mean CPU Time			
1	10	1	1	982.6	5.595	1036.	4.681	1041.	2.963	0.04845	0.00897	58.54
2	10	1	10	393.0	14.39	405.7	10.44	423.9	4.769	0.04047	0.00013	3.222
3	10	1	20	372.4	20.21	378.4	19.09	378.4	19.09	0.14720	0.14720	5.319
4	10	10	1	271.6	8.135	375.0	6.421	383.6	6.799	1.03E-	7.41E-	60.31
5	10	10	10	118.8	14.51	202.5	12.21	204.8	11.61	1.49E-	7.69E-	36.29
6	10	10	20	167.5	27.37	162.9	15.58	156.5	16.74	0.16462	0.18369	-
7	10	20	1	262.3	7.862	306.5	6.707	320.7	5.656	1.12E-	5.21E-	38.29
8	10	20	10	161.0	15.98	207.0	11.20	204.4	12.61	6.7E-22	2.16E-	9.616
9	10	20	20	146.9	27.08	163.7	17.38	157.7	18.48	4.09E-	0.00096	1.725
10	50	1	1	54371	158.2	56149	110.6	71486	24.25	0.19442	2.09E-	37.31
11	50	1	10	13166	608.8	13217	408.7	16733	109.9	0.46048	2.12E-	0.252
12	50	1	20	9744.	971.5	10774	661.5	11740	223.4	0.00045	1.17E-	3.323
13	50	10	1	14639	150.4	16553	85.53	18558	25.15	0.00061	5.08E-	29.49
14	50	10	10	4510.	405.7	4898.	347.8	5761.	95.45	6.67E-	4.03E-	6.706
15	50	10	20	3067.	817.1	3358.	580.3	3684.	188.8	4.66E-	2.76E-	1.231

16	50	20	1	9721.	120.8	10932	100.4	12334	39.87	0.00094	1.14E-	59.34
17	50	20	10	3603.	505.4	3965.	339.4	4413.	95.00	7.03E-	2.52E-	2.179
18	50	20	20	2269.	794.6	2371.	682.3	2601.	252.1	0.00372	8.33E-	0.905
19	10	1	1	49406	294.0	52583	249.8	55922	112.7	0.00021	9.97E-	718.9
20	10	1	10	11682	1452.	12814	1162.	13201	484.3	1.38E-	4.57E-	39.10
21	10	1	20	62417	3082.	67361	2600.	73592	856.0	5.3E-08	9.13E-	10.26
22	10	10	1	13030	348.8	14607	408.4	15691	70.60	2.4E-06	1.18E-	-
23	10	10	10	32810	1482.	35140	1214.	39327	324.1	5.27E-	6.76E-	8.698
24	10	10	20	15975	3061.	17150	2651.	19450	723.0	2.02E-	8.93E-	2.864
25	10	20	1	80240	340.7	85650	279.3	98316	86.50	0.00080	3.37E-	88.09
26	10	20	10	18772	1629.	19957	1434.	22229	403.5	1.36E-	1.21E-	6.051
27	10	20	20	10497	2978.	11513	2347.	12311	880.3	1.17E-	1.12E-	1.610
											Avrg	35.72

The results demonstrate the superiority of TTH algorithm, compared to other two algorithms. Because compared to OGA, the P-value is lower than 0.05 in 23 cases and compared to $GA_{Ullrich,}$ the P-value is lower than 0.05 in 25 cases, H_0 is rejected. It means that the transfer of chromosomes from one generation to the preceding generations prevents the convergence of the solutions, and the algorithm examines a wider range of solutions. In other words, in the conventional GA, the algorithm usually converges into one solution after few generations, and the chromosomes of the last generations would be similar to each other. Transferring the chromosomes to the previous generations causesproduceing more diverse solutionss and delaying the convergence of the algorithm.

One of the advantages of the TTH algorithm is its escaping mechanism from rapid convergence through its backward operator. The termination criterion and other characteristics (with the exception of the backward mechanism) are the same for algorithms. The TTH escaping mechanism from rapid convergence causes its higher CPU time. To illustrate the economic nature of this amount, the improvement rate indicator was defined as follows:

Improvement Rate = (Result of OGA - Result of TTH)/ (TTH CPU time - OGA CPU time)

This indicator indicates the improvement in the result per each unit of additional CPU time (second). The average improvement rate for all of 27 problems is 35.72745. Additionally, the hardware performance is enhanced nowadays, continuously. In this paper a computer with an Intel Corei5 2.5GHz CPU is used to solve the problems. This ratio could be enhanced if a newer hardware is used.

Comparison of TTH with optimum solutions

To compare TTH results with optimum solutions obtained by CPLEX solver, a number of small size random problems are randomly produced. Each problem is shown by three parameters. The first parameter refers to No, and the two others indicate Ns and Nv, respectively. Other parameters are randomly selected from Table 4. The results show that TTH produces nearly optimal solutions with a lower CPU time than CPLEX.

Table 4. Comparison of TTH and the optimum solutions.

No.	Problem	CPLEX Result	CPLEX CPU time	TTH Result	TTH CPU time
1	3×2×2	771	38	771	6
2	3×3×3	738	45	738	7
3	4×3×4	1097	125.73	1101	6
4	4×4×3	871	150	871	9
5	5×2×2	1742	643.7	1747	16
6	5×3×2	1632	572.7	1632	12
7	5×2×3	1677	634.1	1681	14
8	5×3×4	1129	753.45	1129	16
9	5×4×3	1086	866	1086	18
10	6×2×2	1841	2203	1847	21

Sensitivity analysis

In what follows, a sensitivity analysis is performed on the main parameters of the problem and the proposed algorithm. For this purpose, the three main problem parameters (number of orders, number of vehicles, and number of suppliers) and the three main parameters of the algorithm (GB, R, and Trans_rate) are considered. The value of each parameter is increased, while the values of the other two remain unchanged. Then, the changes in the objective function value and the solving time are measured. To perform sensitivity analysis for the algorithm parameters, one problem is considered that is run by various parameters as follow: number of orders=20, number of vehicles=5, and number of suppliers=5. Tables 5 and 6 show the considered values of the parameters, obtained objective function values, and the solving times.

Table 5. Sensitivity analysis

Problem	Number of vehicles	Number of suppliers	Number of orders	Objective function	CPU Time (Second)
1	1	5	5	100	3.292082
2			10	263	7.586503
3			20	3514	17.212057
4			40	14607	28.639465
5			60	39164	40.995481
6			80	99745.5	87.938393
7			100	231536	92.478736
8			200	512839.5	111.53341
Problem	**Number of vehicles**	**Number of suppliers**	**Number of orders**	**Objective function**	**CPU Time (Second)**
1	1	5	40	14607	28.639465
2	5			9413	54.793950

3	10			4988	68.638237
4	15			3593.5	79.601942
5	20			2919.5	90.431426
6	25			2673	108.587833
7	30			2517.5	130.759344
8	40			2137.5	145.959171
Problem	Number of vehicles	Number of suppliers	Number of orders	Objective function	CPU Time (Second)
1	10	5	20	991.5	60.378096
2		10		723.5	147.699314
3		15		565.5	281.844315
4		20		530	382.662983
5		25		464.5	559.234370
6		30		439.5	734.859380
7		40		421	885.120670
8		50		393	1424.923936

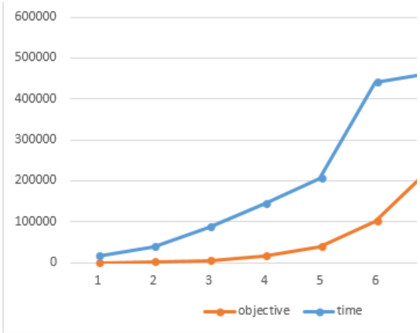
Table 6. Sensitivity analysis of the three main parameters of the algorithm

Test number	GB	R	Trans_rate	Objective function	CPU Time (Second)	Best objective function
1	10	5	0.1	1389	40.95174	1264.5
2			0.2	1317.5	43.86302	
3			0.3	1332	35.75518	
4			0.4	1265.5	50.75994	
5			0.5	1264.5	61.24447	
6			0.6	1313.5	61.28478	
7			0.7	1310.5	42.11245	
8			0.8	1322.5	49.77619	
9			0.9	1330	58.12223	
Test number	GB	R	Trans_rate	Objective function	CPU Time (Second)	Best objective function
1	6	5	0.2	1243.5	53.5903	1243.5
2	8			1332.5	47.32724	
3	10			1317.5	43.86302	
4	12			1355.5	38.88262	
5	14			1399.5	28.36214	
6	16			1337	32.75351	
7	18			1515.5	28.53043	
8	20			1561	27.71693	
9	25			1561	29.10276	
Test number	GB	R	Trans_rate	Objective function	CPU Time (Second)	Best objective function
1	10	1	0.2	1365	45.84586	1305.5

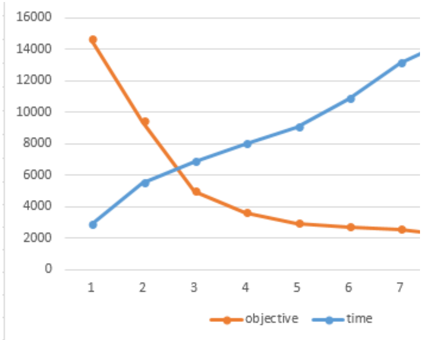
2	2	1361	41.47832
3	3	1354	40.48171
4	4	1342.5	39.45327
5	5	1317.5	43.86302
6	6	1311.5	47.93568
7	7	1306	50.5726
8	8	1305.5	49.10202
9	9	1322.5	39.62954

The considered parameters of GB, R and Trans_rate are related to TTH. Solving the test problem by OGA gives an objective function equal to 1495 with a CPU time equal to 26.382593 seconds. The result of OGA is better than the results of the TTH algorithm in 3 cases only when the value of GB is high.

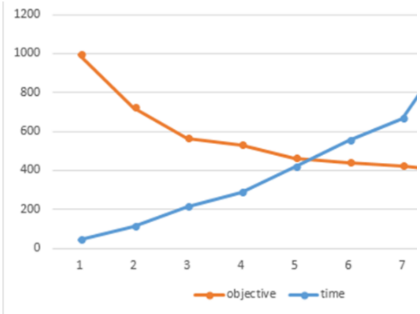
Figure 3 shows the changes in the objective function value and solving time, when Trans_rate, GB, R, the number of orders, vehicles and suppliers are changed. Figure 3(a) illustrates that by increasing the number of orders, the objective function value and the solving time of the algorithm are also increased. Figure 3(b) and Figure 3(c) show that increasing the number of vehicles and suppliers reduces the objective function value, while increasing the solving time. Figure 3(d) illustrates that increasing Trans_rate causes different behavior on results and CPU time. But the figure show that in cases that CPU time is high, the objective function is low and vice versa. Figure 3(e) represents that increasing GB reduces the CPU time of the algorithm, while increasing the objective function value. Figure 3(f) shows that increasing R decreases the objective function firstly and then increases it. The sensitivity analysis shows that the best answer is given when the Trans_rate is medium (0.5), GB is low and R is approximately high. When Trans_rate is low, no experiment and knowledge from current chromosomes structures are conveyed to the previous generation and the performance of time traveling process is reduced. When Trans_rate is high, almost all the current chromosomes are conveyed to the past and no changes in population is occurred by time travelling process. On the other hand, if GB is high, then the algorithm maybe be converged to a local optimum and so, the time travelling process may convey some local optimum solutions to the past. These local optimum solutions reduce the performance of the algorithm. Finally, when R is low, there are more similarities between the current and the past generations. Due to these similarities, the conveyance of the chromosomes during time traveling process has not made a significant change in the population structure.



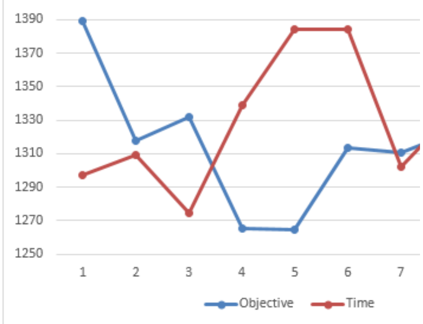
(a) Changes in the number of orders



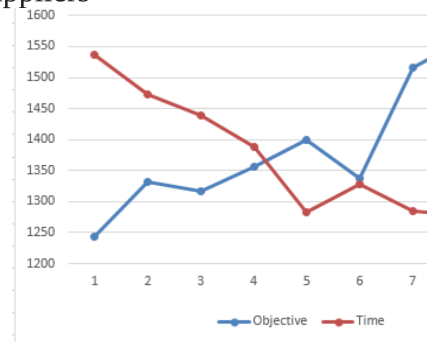
(b) Changes in the number of vehicles



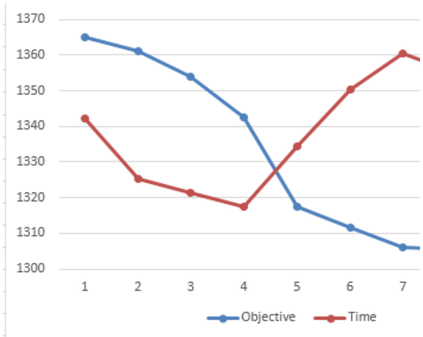
(c) Changes in the number of suppliers



(d) Changes in Trans_rate



(e) Changes in GB



(f) Changes in R

Figure 3. Sensitivity analysis results.

Discussion and Conclusion

The comparison results show that TTH algorithm performs better than OGA. The structure of OGA is quite similar to that of TTH algorithm. The only difference is that OGA has no time travel mechanism. In other words, in OGA, GB equals a large positive number.

The reason for the superiority of TTH algorithm to OGA is that in the conventional GA, the algorithm usually converges into one solution after some generations, and the chromosomes of the last generation would be similar to each other. Therefore, when the crossover operation is performed on them, the offspring chromosomes are also similar and the algorithm would not be able to search for new solutions. Hence, further iterations of the algorithm would not lead to a considerable improvement in the current best solution. The time travel mechanism allows for transferring the chromosomes to the previous generations, where the chromosomes may produce more diverse offspring. This would delay the convergence of the algorithm and enhance the chance that more areas of the solution space are searched.

Moreover, the results of the sensitivity analysis show that an increase in the number of orders results in the increase of the objective function value. The reason is that by increasing the total number of orders, the suppliers are faced with more orders that should be processed and the vehicles have more orders for delivery. In case this increase in the orders is not a much pleasant scenario for the managers, the orders may be outsourced in order to have less orders for scheduling and reduce the tardiness and earliness of the orders.

The results also suggest that the objective function is reduced when the total number of suppliers and vehicles are increased. This is because there are more suppliers and vehicles for orders to be assigned to. Thus, the workload is reduced, and less time is spent on processing and delivering orders. Therefore, the managers may use more suppliers and vehicles in order to reduce the objective function value. It should be mentioned that excessive increase in the number of suppliers and vehicles would lead to negative consequences to the supply chain. Therefore, the managers should keep a balance between the objective function value and the number of the suppliers and vehicles in order to achieve the desired results.

Moreover, by increasing GB, the objective function value is increased and the TTH algorithm will resemble OGA. The results also suggest that the objective function is reduced when R is increased. As the chromosomes are transferred to farther generations and are more differentiated, numerous offspring are produced and this prevents premature convergence. Furthermore, the selection of Trans_rate values also influences the performance of the algorithm, since the transfer of low chromosomes to the preceding generations turns the TTH algorithm into OGA and their high transmission makes the solution convergent. Therefore, selecting the appropriate level of these three parameters has a positive effect on the performance of the algorithm.

For the future research, the problem may be studied when more objective functions are considered, such as reducing pollution and decreasing transportation costs. Combining TTH algorithm with other heuristics and meta-heuristics, including simulated annealing and bee colony, is another subject for future studies.

References

Beheshtinia, M. A., & Ghasemi, A. (2017). A multi-objective and integrated model for supply chain scheduling optimization in a multi-site manufacturing system. *Engineering Optimization*, 50(9), 1415-1433.

Beheshtinia, M. A., Ghasemi, A., & Farokhnia, M. (2017). Supply chain scheduling and routing in multi-site manufacturing system (case study: a drug manufacturing company). *Journal of Modelling in Management*, 13(1), 27-49.

Borumand, A., & Beheshtinia, M. A. (2018). A developed genetic algorithm for solving the multi-objective supply chain scheduling problem. *Kybernetes*, 47(7), 1401-1419.

Chang, Y. C., & Lee, C. Y. (2004). Machine scheduling with job delivery coordination. *European Journal of Operational Research*, 158(2), 470-487.

Chang, Y.C., Chang, K.H., & Kang, T.C. (2015). Applied variable neighborhood search-based approach to solve two-stage supply chain scheduling problems. Journal of Testing and Evaluation, 44(3), 1337-1349

Fahimnia, B., Luong, L., & Marian, R. (2012). Genetic algorithm optimisation of an integrated aggregate production–distribution plan in supply chains. *International Journal of Production Research*, 50(1), 81-96.

Han, B., & Zhang, W. J. (2015). On-line Supply Chain Scheduling Problem with Capacity Limited Vehicles. *IFAC-PapersOnLine*, 48(3), 1539-1544.

Holland, J. H. (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press.

Karaoğlan, İ., & Kesen, S. E. (2017). The coordinated production and transportation scheduling problem with a time-sensitive product: a branch-and-cut algorithm. *International Journal of Production Research*, 55(2), 536-557.

Kumar, M., Vrat, P., & Shankar, R. (2004). A fuzzy goal programming approach for vendor selection problem in a supply chain. *Computers & Industrial Engineering*, 46(1), 69-85.

Low, C., Chang, C. M., & Gao, B. Y. (2017). Integration of production scheduling and delivery in two echelon supply chain. *International Journal of Systems Science: Operations & Logistics*, 4(2), 122-134.

Moon, C., Lee, Y. H., Jeong, C. S., & Yun, Y. (2008). Integrated process planning and scheduling in a supply chain. *Computers & Industrial Engineering*, 54(4), 1048-1061.

Selvarajah, E., & Zhang, R. (2014). Supply chain scheduling at the manufacturer to minimize inventory holding and delivery costs. *International Journal of Production Economics*, *147*, 117-124.

Ullrich, C. A. (2013). Integrated machine scheduling and vehicle routing with time windows. *European Journal of Operational Research*, 227(1), 152-165.

Xu, S., Liu, Y., & Chen, M. (2017). Optimisation of partial collaborative transportation scheduling in supply chain management with 3PL using ACO. *Expert Systems with Applications*, 71, 173-191.

Yeung, W. K., Choi, T. M., & Cheng, T. C. E. (2011). Supply chain scheduling and coordination with dual delivery modes and inventory storage cost. *International Journal of Production Economics*, 132(2), 223-229.

Yimer, A. D., & Demirli, K. (2010). A genetic approach to two-phase optimization of dynamic supply chain scheduling. *Computers & Industrial Engineering*, 58(3), 411-422.

Yin, P. Y., Lyu, S. R., & Chuang, Y. L. (2016). Cooperative coevolutionary approach for integrated vehicle routing and scheduling using cross-dock buffering. *Engineering Applications of Artificial Intelligence*, 52, 40-53.

Zegordi, S. H., & Beheshti Nia, M. A. (2009). Integrating production and transportation scheduling in a two-stage supply chain considering order assignment. *The International Journal of Advanced Manufacturing Technology*, 44(9), 928-939.

Zegordi, S. H., & BeheshtiNia, M. A. (2009). Integrating production andtransportation scheduling in a two-stage supply chain considering order assignment. *The International Journal of Advanced Manufacturing Technology*, 44(9-10), 928-939.