



Coordinating Order Acceptance and Batch Delivery for an Integrated Supply Chain Scheduling

A. Noroozi^a, M. Mahdavi Mazdeh^{*a}, M. Rasti-Barzoki^b

^a Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran

^b Department of Industrial and Systems Engineering, Isfahan University of Technology, Isfahan, Iran

PAPER INFO

Paper history:

Received 12 January 2017

Received in revised form 17 February 2017

Accepted 10 March 2017

Keywords:

Order Acceptance

Integrated Production-distribution

Batch Delivery

Genetic Algorithm

Particle Swarm Optimization

ABSTRACT

This paper develops Order Acceptance for an Integrated Production-Distribution Problem in which Batch Delivery is implemented. The aim of this problem is to coordinate: (1) rejecting some of the orders (2) production scheduling of the accepted orders and (3) batch delivery to maximize Total Net Profit. A Mixed Integer Programming is proposed for the problem. In addition, a hybrid meta-heuristic algorithm is developed. For a quick exploration around a solution, a Local search is proposed. Two simple heuristics for initial population and a heuristic for batching are proposed. Besides, data is generated to evaluate the performance of algorithms and compare with each other based on comprehensive experiments.

doi: 10.5829/idosi.ije.2017.30.05b.10

1. INTRODUCTION

Classical *scheduling* just focused on the determination of orders schedules for production without taking into account distribution such as [1, 2]. However, to achieve optimal scheduling performance of a supply chain, it is critical to integrate production and distribution [3]. Vroblefski et al. [4] have highlighted that one of the main costs in distribution is the cost of transportation. One of the most important approaches that can decrease the distribution costs is *Batch Delivery* (BD). BD is defined as the process of batching the orders and delivering the batches using transporters [5]. Therefore, scheduling of an *integrated production and distribution supply chain* with the implementation of batch delivery can be effective for a supply chain to achieve the optimal goal. Chen [3] presented a survey of such existing models. For extending the due date management this field. Yin et al. [6] studied a BD single machine scheduling to minimize earliness, tardiness,

holding time, window location, and window size. Rasti-Barzoki and Hejazi [7] considered the same problem to minimize the weighted number of tardy jobs, due date assignment cost and batch set up cost. In another work, Rasti-Barzoki and Hejazi [8] studied the same problem considering the resource allocation. Considering the delivery cost, and investigating minimizing the sum of weighted flow times for BD and single machine scheduling, Mazdeh et al. [9] presented structural properties of the problem. As it can be seen in the BD literature, there is one important gap. For logistics activities and BD, one of the most significant resources in distribution stage is transporter [11, 12]. In which the articles supposed that there are sufficient transporters however, many of the distribution centers have not the enough transporters to deliver the batches to customers and a batch has to wait until a transporter return to the company and is being free to deliver another batch. This gap is addressed in this paper and hereafter, this condition is named as *Round Trip Transportation*. Furthermore, the papers supposed that the orders have same size and the capacity of the transporters is unlimited. However, in many industries such as food

*Corresponding Author's Email: mazdeh@iust.ac.ir (M. Mahdavi Mazdeh)

and beverage, the customers have the different sizes of the pallets or boxes of the products as well as a transporter can deliver a constraint volume of the orders. So, this paper considers a practical aspect of the BD, that is, each order occupies different physical space in the transporter in which the transporters have a constraint capacity. On the other hand, in most of the companies, there are not sufficient capacities to production and distribution all of the orders. While, "Accepting orders without considering their possible costly impact on capacity can mean the firm is paying for the privilege or profitability of accepting an order" [13]. In addition, in the supply chain, the sales department tends to accept orders as much as possible to maximize revenue, while, due to available capacity and resources, the production and distribution parties tend to reject orders as much as possible to minimize cost. The tradeoff among orders' revenue and production-distribution costs is achievable with *Order Acceptance and scheduling* (OAS) approach. In the OAS approach, all of the brought orders are not accepted, in which some of the orders can be rejected. Slotnick [14] presented a taxonomy and review of OAS literature.

Nobibon and Leus [15] studied OAS problem in a single machine environment. Og et al. [16] studied OAS in this environment with considering the orders with release dates, due dates, deadlines, processing times, sequence dependent setup times and revenues. Cesaret et al. [17] solved OAS on a single machine in which the orders have release dates and sequence dependent setup times. Maximizing the total net revenue (TNP) of the orders in a two-machine flow shop with implementing OAS approach is studied by Wang et al. [18]. They formulated the problem as mixed-integer programming models. For this problem, Esmailbeigi et al. [19] presented two mixed integer programming model. Lin and Ying [20] proposed a simulated annealing algorithm for OAS in a permutation flow shop problem. For this problem, Rahman et al. [21] proposed a simple heuristic to accept orders and a GA for scheduling the accepted orders; and Lei and Guo [22] studied a multi-objective approach for minimizing makespan and maximizing TNP. A mixed integer linear programming model is proposed in their work. For OAS, another mixed integer programming model is presented by Emami et al. [23] in which their scheduling environment was non-identical parallel machines. An m parallel machine environment is studied by Thevenin et al. [25].

As it can be seen in the literature, the entire studied problem in OAS, investigated only production and, to the best our knowledge, so far, any paper has not studied OAS in distribution or integrated production and distribution. Therefore, in addition to the mentioned gaps, this gap is also addressed in this paper. This paper studied the OAS problem in an integrated production and distribution.

Furthermore, since in many industries, each customer has his own orders and the company ships each batch to the customer by a vehicle separately, this paper supposes that there are multiple customers. In this condition, a batch cannot contain the orders of two or more customers. As well as, due to that the permutation flow shop is one of the well-known kinds of production environment with many practical applications in manufacturing systems [26], this paper studied the problem in a flow shop production line that has m machine.

We proposed a mixed integer programming model for the problem. Furthermore, we provide a heuristic to form the batches. Then, utilizing the heuristic and developing Genetic Algorithm (GA) and Particle Swarm Optimization (PSO), a hybrid Meta-heuristic algorithm is proposed. In proposed hybrid algorithm, a local search is proposed. Moreover, two heuristics are proposed and analyzed as an initial solution of the hybrid algorithm.

The remainder of the paper is organized as follows. In section 2, after describing the problem in detail, a mixed integer programming model is proposed. Section 3 presents the different parts of the developed algorithm and section 4 introduces the data generation. Section 5 presents the investigating of the performance of the algorithms, proposed heuristics and local search. Finally, Section 6 concludes the paper.

2. PROBLEM FORMULATION

2. 1. Problem Statement

In the considered integrated supply chain, there is a flow shop production line, a distribution center with a constant number of transporters and a number of customers. A pool of the orders arrives to the sales department. Subsequently, due to the production and distribution cost, capacity, tardiness cost, and revenue, a number of the orders are accepted and the rests are rejected. The accepted orders are scheduled on the flow shop in the production stage. In the distribution, the processed orders are batched and shipped directly to the customer using transporters. In batch forming, each order occupies an individual physical space in the transporter and the capacity of the transporter is considered the maximum accessible space of a batch. In each batch, only the orders of a customer must be placed. For shipping, according to the Round Trip Transportation, when a batch is formed, if there is not a free transporter, the batch has to wait until a transporter return to the company. In addition, to the delivery cost, after delivery, if it is delivered later than the due date, the company has to pay a tardiness cost. The objective is to maximize Total Net Profit (TNP) resulting from the difference between revenue of the accepted orders and the cost of the transportation and tardiness.

2. 2. Problem Assumptions

- All jobs are available at time zero.
- The parameters are known and deterministic.
- Machine and transporter breakdown is not allowed.
- The buffer between the two machines is infinite.

2. 3. Notations

Index list

- k The number of customers ($k=1, \dots, K$)
- j The number of orders ($j=1, \dots, n_k$)
- $N = \sum_{k=1}^N n_k$ Total number of orders
- b The batch number ($b = 1, \dots, N$)
- p Order sequence position in schedule ($p = 1, \dots, N$)
- v The vehicle number ($v = 1, \dots, V$)

Parameters list

- V The total number of vehicles
- π_{jk} The revenue of the order j of customer k
- β_{jk} Tardiness cost of the order j of customer k
- θ_k Transportation cost of the customer k
- p_{jk} Process time the order j of customer k
- s_{jk} Occupied space by the order j of customer k
- Cap Vehicle loading capacity
- t_k Transportation time of customer k
- d_{jk} Due date of the order j of customer k

Decision variables list

- $x_{jk,p}$ 1: if order j customer k place in position p in sequence
0: otherwise
- y_{jk} 1: if order j customer k is accepted
0: otherwise
- C_{pm} The completion time of the order in the position p on machine m
- C_{jk} The completion time the order j of customer k
- $A_{jk,b}$ 1: if the order j of customer k is allocated to the batch b
0: otherwise
- R_{bk} The ready time to the batch b of the customer k
- $q_{bk,v}$ 1: if the batch b of the customer k is transported by vehicle v
0: otherwise
- U_{bk} 1: if the batch b is allocated to customer k
0: otherwise
- $\delta_{v,i}$ Delivery time of batch i th in the shipping sequence which is transported by the vehicle v
- $D_{b,k}$ Delivery time of the batch b
- d'_{jk} Delivery time of the order j of customer k
- T_{jk} Tardiness of the order j of customer k

2. 4. Mathematical Model

$$Max\ profit = \sum_{k=1}^K \sum_{j=1}^{n_k} \pi_{jk} y_{jk} - \left(\sum_{k=1}^K \sum_{j=1}^{n_k} \beta_{jk} T_{jk} + \sum_{k=1}^K \sum_{b=1}^N \sum_{v=1}^V \theta_k q_{kb,v} \right) \quad (1)$$

Subject To:

$$\sum_{p=1}^N x_{jk,p} = y_{jk} \quad ; \quad k=1, \dots, K \quad j=1, \dots, n_k \quad (2)$$

$$\sum_{k=1}^K \sum_{j=1}^{n_k} x_{jk,p} \leq 1 \quad ; \quad p = 1, \dots, N \quad (3)$$

$$C_{1m} = \sum_{i=1}^m \sum_{k=1}^K \sum_{j=1}^{n_k} x_{jk,p} P_{jk,i} \quad ; \quad m = 1, \dots, M \quad (4)$$

$$C_{pm} \geq C_{p-1m} + \sum_{k=1}^K \sum_{j=1}^{n_k} x_{jk,p} P_{jk,i} \quad ; \quad m = 1, \dots, M \quad p = 2, \dots, N \quad (5)$$

$$C_{pm} \geq C_{p-1m} + \sum_{k=1}^K \sum_{j=1}^{n_k} x_{jk,p} P_{jk,i} \quad ; \quad m = 1, \dots, M \quad p = 2, \dots, N \quad (6)$$

$$C_{jk} = \sum_{k=1}^K \sum_{j=1}^{n_k} C_{pm} x_{jk,p} \quad ; \quad k=1, \dots, K \quad j=1, \dots, n_k \quad (7)$$

$$\sum_{k=1}^K U_{bk} \leq 1 \quad ; \quad b = 1, \dots, N \quad (8)$$

$$\sum_{b=1}^N U_{bk} \leq n_k \quad ; \quad k=1, \dots, K \quad (9)$$

$$\sum_{b=1}^N A_{jk,b} \leq y_{jk} \quad ; \quad k=1, \dots, K \quad b = 1, \dots, N \quad (10)$$

$$\sum_{j=1}^{n_k} S_{jk} A_{jk,b} \leq Cap \cdot U_{bk} \quad ; \quad k=1, \dots, K \quad j=1, \dots, n_k \quad (11)$$

$$R_{bk} \geq C_{jk} A_{jk,b} \quad ; \quad k=1, \dots, K \quad j=1, \dots, n_k \quad b = 1, \dots, N \quad (12)$$

$$\sum_{k=1}^K \sum_{b=1}^N q_{bk,v} = 1 \quad ; \quad v = 1, \dots, V \quad (13)$$

$$\sum_{v=1}^V q_{bk,v} = 1 \quad ; \quad k=1, \dots, K \quad b = 1, \dots, N \quad (14)$$

$$\delta_{v,0} = 0 \quad ; \quad v = 1, \dots, V \quad (15)$$

$$\delta_{v,i} = \delta_{v,i-1} + \sum_{k=1}^K \sum_{b=1}^N \sum_{v=1}^V (2t_k + R_{k,b}) q_{kb,v} \quad ; \quad i = 1, \dots, N \quad v = 1, \dots, V \quad (16)$$

$$D_{bk} = \sum_{v=1}^V \sum_{i=1}^N \delta_{v,i} q_{bk,v} \quad ; \quad k=1, \dots, K \quad b = 1, \dots, N \quad (17)$$

$$d_{jk} = \sum_{b=1}^N D_{bk} A_{jk,b} \quad \begin{matrix} j=1,\dots,n_k \\ k=1,\dots,K \end{matrix} \quad (18)$$

$$T_{jk} \geq d_{jk} - d_{jk} \quad ; \quad \begin{matrix} j=1,\dots,n_k \\ k=1,\dots,K \end{matrix} \quad (19)$$

$$T_{jk} \geq 0 \quad ; \quad \begin{matrix} k=1,\dots,K \\ j=1,\dots,n_k \end{matrix} \quad (20)$$

$$\begin{matrix} x_{jk,p}, y_{jk}, A_{jk,b}, q_{bk,v} \in \{0,1\} \\ C_{jk}, R_{bk}, \delta_{v,i}, D_{bk}, d_{jk}, T_{jk} \geq 0 \\ \begin{matrix} k=1,\dots,K \\ j=1,\dots,n_k \\ p=1,\dots,N \\ b=1,\dots,N \\ v=1,\dots,V \end{matrix} \end{matrix} \quad (21)$$

In the model, the objective function (1) maximizes the TNP resulting from the difference between revenue of the accepted orders and the cost of the transportation and tardiness. In Equation (2), if an order is accepted, it is assigned to a position in the production sequence. To guarantee that each position is assigned only to one order, constraint (3) assigns each position to one accepted order. Constraint (4) computes the completion time of the first order in the sequence, on each machine. Constraints (5) and (6) are used to compute the completion time of the order in position p th on the machine m th. Finally, constraint (7) computes the completion time of the order j of customer k on the last machine (M). In the problem, there are at most N batches that the orders of a same customer must be placed in each batch. To this purpose, constraint (8) assigns each batch to one customer and constraint (9) guarantees that the maximum number of the allocated batch to the customer k should be at most n_k batches.

Based on constraints (8) and (9), Equation (11) assigns each accepted order to one batch and constraint (10) guarantees that the occupied space of the orders in the batches, is smaller than the transporter's capacity. In the shipping, a batch is ready, when all of the orders are prepared for shipping. That is, a batch is transported when the production processing of the orders in the batch are completed. These conditions are considered in constraint (12). After that, Equations (13) and (14) allocate one transporter to one formed batch to ship directly to the customers. The delivery time of the batches in the transporting sequence i th that is transported by transporter v , is computed by Equations (15) and (16). Using these delivery times, Equation (17) computes the delivery time of the batch b of customer k . Based on Equation (17), Equation (18) computes the delivery time of the order j of the customer k that were placed in the batch b of customer k . Finally, the tardiness of each order is computed by constraints (19) and (20). Constraints (21) show the decision variables.

3. SOLUTION APPROACH

BD is strongly NP-hard [3]. As well as, Ghosh [27] showed that order acceptance and scheduling with lateness penalty is NP-hard [28]. Therefore, our considered problem is NP-hard. So, employing meta-heuristic algorithms are preferable. The GA as well as PSO has been popular in academia and the industry mainly because of its intuitiveness, ease of implementation, and the ability to effectively solve highly nonlinear mixed integer optimization problems. Therefore, in this paper, a hybrid meta-heuristic algorithm has been developed in which GA is used to encode the order acceptance approach and PSO is implemented to schedule the orders and batches.

3.1. Genetic Algorithm

To encode the OAS problem, the good ideas are presented in the literature [17, 20, 29, 30], but, to the best of our knowledge, there are two gaps:

(1) In the search process, an order may be removed early; however, if the order is removed later, it may improve the TNP. In their work, to preventing of the losing an improvement, a local search or other approaches were not proposed. (2) The base of the most of them is the scheduling of the accepted order. However, accept or reject an order is usually performed randomly. Therefore, an encoding scheme and search mechanism must be strong to investigate a high volume of accept or reject of an order. To this purpose, due to the efficiency of GA to handle the zero-one problems using the binary coding, we implement the acceptance or rejection decisions strongly using GA. In this coding, '0' represents rejecting and '1' represents accepting an order by a chromosome (Figure 1).

In GA, after defining a representation code, a set of chromosomes are created as the initial population. To update the generation, some of the chromosomes are selected based on the fitness function. In the problem, after accepting some of the orders, the fitness function is depended on the scheduling, batching and shipping. In our proposed algorithm, this process is performed by the PSO algorithm. After this process, genetic operators (crossover and mutation) generate the new chromosomes (offspring).

3.2. Particle Swarm Optimization

PSO is an efficient algorithm dealing with optimization problems [31, 32]. In the developed PSO, each *particle* represents a possible schedule of the accepted orders. The PSO algorithm is inherently continuous. In this study, using the smallest position value (SPV), each continuous value of the PSO is transformed into the acceptable discrete value which is the position of the order in the schedule. This method is presented in Figure 2. In this figure, based on SPV, the smallest particle value is

$x_{i5}^t = -1.20$. Therefore, the dimension number, i.e. 5, is selected as the order that place in the first particle position and so on.

In the proposed algorithm, after determining the schedule of the accepted orders, a heuristic is applied to form batches and BD scheduling.

3. 3. Heuristic Algorithm In order to batch the orders and determining the shipping sequence of the batches, a heuristic algorithm is presented, which is based on the order ready time; namely *rFF-H*. After determining the schedule of the accepted orders by SPV rule, the completion time of the orders (c_{jk}) is equal to the order ready time for delivery (r_{jk}).

$$c_{jk} = r_{jk} \tag{2}$$

Consequently, at the distribution stage, there are the accepted orders with the ready time, due date, transportation cost and tardiness cost that should be appropriately formed the batches and transported to customers. Mehta and Uzsoy [33] developed the batch apparent tardiness cost algorithm, adapting the apparent tardiness cost heuristic of Morton and Vepsalainen [34].

We adopt their heuristic to our problem with modifying:

- (1) Sort all of the accepted orders of each customer in non-decreasing order of ready time and nondecreasing order of weighted due date for orders with the same ready times.

After determining of the orders' schedule for batching another heuristic (*H*) is used to form the batches.

- (2) Heuristic *H*. For the customer *k*, the first order in the schedule is placed in batch 1 of the customer. For the second order, if the size of the order in the schedule is not larger than the remaining capacity of the batch, the order is placed in the batch. Otherwise, this batch is closed. This process is implemented for the next order in the schedule. For all of the customers, this process is repeated until all of orders are placed in batches.

Subsequently, it must be determined the shipping sequence of the batches which highly depends on its ready time. The batch shipping sequence is determined with the following step:

- (3) Sort the batches based on the descending order of their ready time.

Binary variabl 1 0 1 1 0 1 1 1 0 1 0 1

Figure 1. Representation of OA

Order number	1	2	3	4	5	6
x_{ij}^t	1.80	-0.99	3.01	-0.72	-1.20	2.15
The permutation of the SPV rule	5	2	4	1	6	3

Figure 2. The SPV method

- (4) The first *V* batches are firstly shipped.
- (5) For other batches, the batch with minimum ready time is shipped with a free transporter which is returned earliest of them and so on.

Using this approach, delivery time of the batch in the position *i*th of sequence is equal to the maximum of the ready time of this batch and returned time of the transporter, plus transportation time.

3. 4. Initial population and updating the velocity and position

To create the initial random solutions, the initial positions and velocities, Equations (24) and (25) are used as following:

$$x_{ij}^0 = x_{\min} + (x_{\max} - x_{\min}) \times r_1 \tag{24}$$

$$v_{ij}^0 = v_{\min} + (v_{\max} - v_{\min}) \times r_2 \tag{25}$$

However, a significant factor on the final result's quality of a search procedure is the initial solution. It has already been recognized and emphasized by many researchers in the recent years [35]. To this purpose, we propose two heuristics for initial population. As it was mentioned, the order completion time of an order is equal to the ready time of the order for batching and transporting. Therefore, a smaller completion time is led to a smaller ready time [36]. In addition, the due date of the orders directly influences on scheduling of the orders and eventually, tardiness of the orders. On the other hand, in the literature, the shortest processing time (SPT) is proposed to minimize the completion time and the earliest due date rule is proposed to minimize the orders lateness (a function of due date) [36]. These rules determine a production schedule of orders. Hence, we must batch the orders according to the schedule. For this purpose, we use the heuristic *H*. According to SPT, EDD and *H*, we proposed two heuristics: *SPT-H* and *EDD-H*. The steps of the heuristics are as following:

SPT-H: (1) schedule the orders based on SPT rule, (2) batch the schedule orders using *H*. (3) ship the batches according to the batch ready time.

EDD-H: (1) schedule the orders based on EDD rule, (2) batch the schedule orders using *H*. (3) ship the according to the batch ready time.

Two initial solutions are created using *SPT-H* and *EDD-H*. Other solutions are generated randomly.

3. 5. Updating the velocity and position

The particles are updated according to the best positions of each particle (*Pbest*) and the position of the best particles is the global optimum (*Gbest*). The main part of PSO is updating the velocity and position of the particles using the following equations:

$$v_{ij}^{t+1} = \omega v_{ij}^t + C_1 r_1^t (Gbest - x_{ij}^t) + C_2 r_2^t (Pbest - x_{ij}^t) \tag{26}$$

$$x_{ij}^{t+1} = x_{ij}^t + v_{ij}^{t+1} \tag{27}$$

In Equation (26), the velocity vector of each particle is updated according to its velocity at the previous stage (v_{ij}^t). Moreover, r_{1j}^t and r_{2j}^t are two random numbers with a uniform distribution in range (0, 1), which are generated independently. Values C_1 and C_2 are learning coefficients (or acceleration coefficients) and control the effect of $Pbest$ and $Gbest$ on the search process. Besides, w indicates the inertia weight coefficient which controls the impact of the previous velocities on the current velocity and χ is the contraction coefficient and insures the convergence of the algorithm. After updating the particles' velocity, their positions (x_{ij}^t) are updated using Equation (27).

3. 6. Local Search In this work, a local search is performed on the PSO in each generation. If this new solution results in a better TNP of accepted orders, the current solution is replaced by the new solution. The process of the local search for current solution terminates if we observe any improvement. In each solution after the batches forming according to the *rFF-H*, the batches are scheduled based on the ready time. For run a local search, the local search schedules the orders' production processing based on their batch ready time. If this solution is better than the best solution, it is replaced with the best solution. An example of applying the local search is demonstrated in Figure 3 according to the data of Table 1.

The pseudo code of the GA_PSO-LS is presented in Figure 4. In this pseudo code, Pr is the reproduction rate. In GA, Pr% of the population is copied to the next

generation and the crossover operator is applied on (1-Pr) % of the population to generate new solutions. Also, Pm is the probability of the mutation. After a new solution is generated by crossover, a random number is generated. If the random number is less than Pm, the mutation operator is applied.

4. DATA GENERATION

In order to evaluate the performance of the proposed algorithms, different sizes of the test problems (small and large) are needed. Four small datasets and five large datasets are created. For determining the number of machines in the flow shop, the number of customer and the number of orders in large size of problems, five combinations from four levels for the number of machines i.e. $M = \{5, 10, 15, 20\}$, three levels for the number of machines i. e. $M = \{5, 10, 15, 20\}$ three levels for the number of machines i.e. $K = \{5, 10, 15\}$ and three levels for the number of orders i.e. $n_k = \{3, 5, 10\}$ are considered:

$M \times K \times n_k = \{5 \times 15 \times 5, 10 \times 30 \times 10, 15 \times 50 \times 10, 15 \times 75 \times 15, 20 \times 100 \times 20\}$. Moreover, for small problems, four combinations from $M = \{2, 3, 4\}$, $K = \{2, 3\}$ and $n_k = \{2, 3, 4, 5\}$ are considered: $M \times K \times n_k = \{2 \times 2 \times 2, 3 \times 2 \times 3, 3 \times 2 \times 5, 4 \times 3 \times 3\}$.

In this paper, the data were generated from a uniform discrete distribution defied in terms of intervals: the revenue on [10,15], the processing time on [1,3], the size of orders on [1,7], the tardiness cost on [3,9] and the transportation cost [5,10]. The vehicle capacity is considered to be 10 for all the test problems. In order to obtain the due date, we propose some equations as Equations (28) to (32).

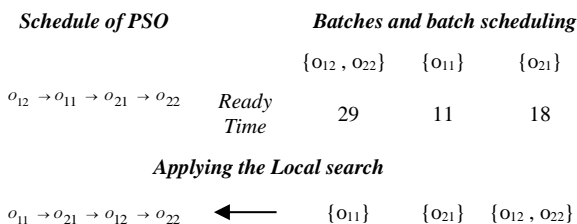


Figure 3. Applying Local Search

TABLE 1. Data for example of Figure 3

Orders	O ₁₁	O ₂₁	O ₁₂	O ₂₂
s_{jk}	6	5	2	7
$p_{jk,1}$	5	6	4	9
$p_{jk,2}$	2	3	2	5

```

START:
Set Parameters;
Generate Initial Population;
Do{ Evaluate Fitness of each solution:
    Set Parameters;
    Generate Initial Population;
    Do{
        Form the batches and BD scheduling by rFF-H
        Evaluate the TNP of New Population;
        Update Pbest;
        Apply Local Search;
        Update velocity;
        Update position;
    }while(stopping criterion is not met);

    Update The Best Chromosome and BestFitness;
    Apply Reproduction (Pr %);
    Apply Selection;
    Apply Crossover (1- Pr %);
    If (rand ≤ Pm) then { apply mutation} ;
}while(stopping criterion is not met);
END;
    
```

Figure 4. The pseudo code of the GA_PSO-LS

$$\bar{B} = BV \times Cap \tag{28}$$

$$\bar{P}_k = OA \times \frac{\sum_{k=1}^K \sum_{j=1}^N P_{jk}}{n_k} \times M \tag{29}$$

$$\bar{S} = OA \times \sum_{k=1}^K \sum_{j=1}^N S_{jk} \tag{30}$$

$$NumTransporters = BV \times \frac{\bar{S}}{B} \tag{31}$$

$$Q_k = \left[\left(\left(\sum_{k=1}^K n_k \right) \times \bar{P} \right) + 2t_k \right] \tag{32}$$

Using *BV* where is the percentage of the occupied space of the transporter, Equation (28) computes the average occupied space of the batch. Since, all of the orders are not accepted, the coefficient *OA* is considered as the percentage of accepted orders and using that, Equation (29) computes the average processing time of each order and Equation (30) presents the total sum of the order sizes. According to the primary experiments, for all the formed batches, *BV*=0.8 and *OA*=0.7. Equation (31) divides the sum of the order sizes by the expected average occupied space of each transporter. Subsequently, *NV* is multiplied by the number of expected vehicles as a coefficient to obtain the number of transporters. Equation (32) computes the average requirement time from the start of the orders processing in production to the final delivery of each batch to the customers. After these calculations above, the due dates are determined as follows.

$$U \sim (L \times Q_k, H \times Q_k) \tag{33}$$

where *L* and *H* are lower and upper limits and are set to be 0.8 and 1.9, respectively. Furthermore, for each combination of large problems, four sample problems are created and for more reliability, each problem is executed ten times. For the proposed algorithms, the stop criteria are as follows:

- (1) Reach a specified number of generations, or
- (2) No change in the TNP in the certain number of repetitions.

According to the experiments, the best parameters of the algorithms are presented in Table 2. All the algorithms were implemented using C# programming language (visual studio 2013) on a computer with a 2.6GHz CPU and a 256Mb RAM.

5. EXPERIMENTAL RESULTS

In order to verify the developed model and evaluate the performance of the algorithms against the exact solution, the commercial solver LINGO 11 is used to

TABLE 2. Parameter values of the algorithms

Parameters	value
(x_{min}, x_{max})	(0, 1)
(v_{min}, v_{max})	(-3.5, 3.5)
Population size of GA	<i>n</i>
Reproduction rate [37]	20%
Probability of mutation (Pm)	0.15
w	1.2
χ	0.9
Population size of PSO	2 × <i>n</i>
C1= C2	1.1

solve the small instances and the outputs are presented in Table 3. The *TNP* and *Time* columns show the objective function and CPU time (millisecond) of the algorithm. In this Table, GA_PSO is the algorithm without the heuristics and local search, GA_PSO-H is the algorithm with the heuristics SPT-H and EDD-H as initial population and without local search, GA_PSO-LS is the algorithm with the local search and without heuristics, and GA_PSO-H-LS is the algorithm with the heuristics as the initial population and local search. As it can be seen in Table 3, the LINGO could find the optimal solution for the two first data sets and, due to the complexity of the problem, it could not reach a solution for other instances, after seven hours computational time. For the four data sets, although the LINGO finds the optimal solution, however, the algorithms can find the near of the optimal solution in a time less than the LINGO. Furthermore, for 3 × 2 × 5 and 4 × 3 × 3 data sets, the algorithms find the optimal or near of the optimal solution and better than the LINGO in a logical time.

Furthermore, we investigate the efficiency of the heuristics in the initial population and local search on the performance of the suggested algorithm. Because the scale of objective functions in each instance and run is different, the relative percent deviation (RPD) is computed for the problems as follows.

$$RPD = \frac{Max_{sol} - Alg_{sol}}{Max_{sol}} \tag{34}$$

where, *Alg_{sol}* is the solution of the algorithm and *Max_{sol}* is the maximum value of the solutions. In this measure, the lowest *RPD* is selected as the best algorithm. The results are shown in Table 4.

As it can be seen in Table 4, GA_PSO-H-LS considerably outperform the GA_PSO, GA_PSO-H, and GA_PSO-LS.

TABLE 3. Comparison of algorithms in small instances

problems	LINGO		GA_PSO		GA_PSO-H		GA_PSO-LS		GA_PSO-H-LS	
	TNP	Time	TNP	Time	TNP	Time	TNP	Time	TNP	Time
2 × 2 × 2	13	12000	13	< 100	13	< 100	13	< 100	13	< 100
3 × 2 × 3	27	619000	26	< 100	26	< 100	27	< 100	27	< 100
3 × 2 × 5	34	> 7 ^h	36	< 1000	37	< 1000	38	< 1000	38	< 1000
4 × 3 × 3	52	> 7 ^h	58	< 2000	61	< 2000	65	< 2000	67	< 2000

Furthermore, Figure 5 shows that the heuristics has a good efficacy on GA_PSO and the proposed local search performs a good exploration around a solution and helps the algorithm to locate in a better neighborhood. Furthermore, to verify the statistical validity of the results, an analysis of variance (ANOVA) is performed. The results show that there is a clear statistically significant difference between performances of the algorithms. The means plot and LSD intervals (at the 95% confidence level) for the algorithms are shown in Figure 6. They do not meet each other and there is no overlap.

We investigate the influence of the heuristics SPT-H and EDD-H, for generating the initial solutions, on the performance of the suggested PSO. To this purpose, we run the large instance 10 × 10 × 5 using the heuristics as some of the individuals in the initial population and do not use the heuristics.

TABLE 4. Average RPD for algorithms

Problem	GA_PSO	GA_PSO-H	GA_PSO-LS	GA_PSO-H-LS
3 × 5 × 3	0.1798	0.1812	0.0553	0.0095
5 × 10 × 3	0.1458	0.1414	0.0334	0.0081
10 × 10 × 5	0.1544	0.1165	0.0756	0.0182
10 × 15 × 5	0.1670	0.1271	0.1035	0.0509
15 × 10 × 10	0.1562	0.1204	0.0943	0.0357
Average	0.1607	0.1373	0.0724	0.0245

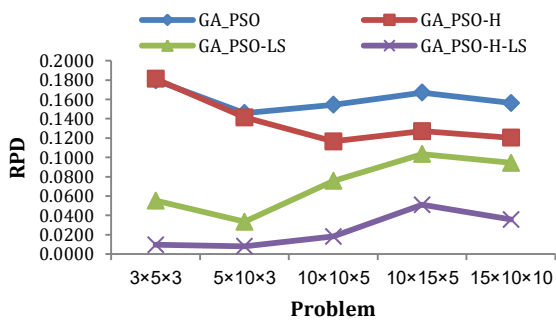


Figure 5. Interaction between algorithm performance (RPD) and size of problems

Figure 7 demonstrated that the proposed heuristics has a considerable effect on the convergence behavior of the algorithm and decreases the number of iterations for achieving to an optimal or near to the optimal solution.

6. CONCLUSION

In this paper, a new approach of the coordination in supply chain: *Order Acceptance, Batch Delivery and Round Trip Transportation* were proposed. The problem includes the integration of the accepting the orders, scheduling, batching, and shipping the batches in which the numbers of transporters are limited.

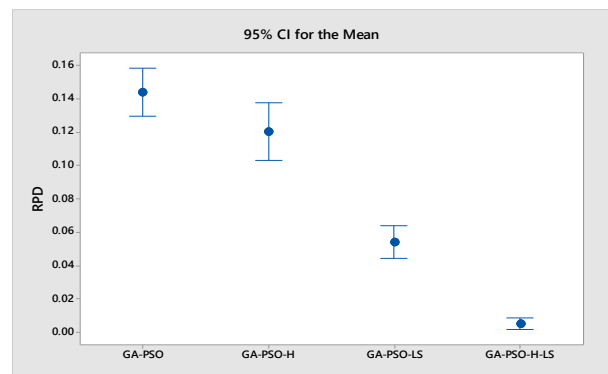


Figure 6. Means plot and LSD intervals for the algorithms

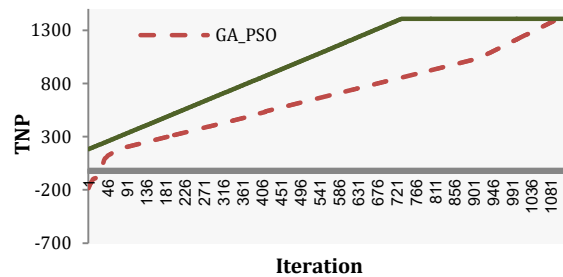


Figure 7. Convergence behavior of the algorithm with and without using the heuristics in initial population

Besides, the most of the previous studies that considered only a single transporter or each transporter exits from the company after delivery a batch, this paper generalized the problem, the paper considered a practical aspect of the problem in which the transporters return to the company to transport other batches, as well as the number of transporters, is limited.

The objective is to maximize the difference between the revenue of accepted orders and the transportation cost and tardiness cost of them, e.g. the company's benefit. To this end, a mixed integer programming model was first proposed. Due to the efficiency of GA to handle the zero-one problems the acceptance or rejection decisions are performed using GA. For the scheduling, batching and shipping the batches of the accepted orders, based on some important properties of the problem, a PSO and a *heuristic* were proposed. In addition to, this paper proposed a *local search* algorithm to improve the quality of the algorithm. A sufficient set of the test problems are generated. In the performed analysis, the appropriate efficiency of GA-PSO-H-LS in solving all the test problems was shown. Moreover, it was revealed that the heuristics in initial population and local search has a good influence on the performance of the algorithm.

For future researches, other transportations and returning mode for transporters can be considered. It is also interesting to apply other heuristics or local search in our proposed algorithms for GA-PSO and developing other met-heuristic algorithms.

7. REFERENCES

- Nahavandi, N., Zegordi, S. and Abbasian, M., "Solving the dynamic job shop scheduling problem using bottleneck and intelligent agents based on genetic algorithm", *International Journal of Engineering-Transactions C: Aspects*, Vol. 29, No. 3, (2016), 347-352.
- Hamta, N., Ghomi, S.F., Bahalke, U. and Golpaigani, H., "Single machine scheduling problem with precedence constraints and deteriorating jobs", *International Journal of Engineering-Transactions A: Basics*, Vol. 24, No. 2, (2011), 115-122.
- Chen, Z.-L., "Integrated production and outbound distribution scheduling: Review and extensions", *Operations Research*, Vol. 58, No. 1, (2010), 130-148.
- Vroblefski, M., Ramesh, R. and Zionts, S., "Efficient lot-sizing under a differential transportation cost structure for serially distributed warehouses", *European Journal of Operational Research*, Vol. 127, No. 3, (2000), 574-593.
- Gao, S., Qi, L. and Lei, L., "Integrated batch production and distribution scheduling with limited vehicle capacity", *International Journal of Production Economics*, Vol. 160, No., (2015), 13-25.
- Yin, Y., Cheng, T., Hsu, C.-J. and Wu, C.-C., "Single-machine batch delivery scheduling with an assignable common due window", *Omega*, Vol. 41, No. 2, (2013), 216-225.
- Rasti-Barzoki, M. and Hejazi, S.R., "Minimizing the weighted number of tardy jobs with due date assignment and capacity-constrained deliveries for multiple customers in supply chains", *European Journal of Operational Research*, Vol. 228, No. 2, (2013), 345-357.
- Rasti-Barzoki, M. and Hejazi, S.R., "Pseudo-polynomial dynamic programming for an integrated due date assignment, resource allocation, production, and distribution scheduling model in supply chain scheduling", *Applied Mathematical Modelling*, Vol. 39, No. 12, (2015), 3280-3289.
- Mazdeh, M.M., Shashaani, S., Ashouri, A. and Hindi, K.S., "Single-machine batch scheduling minimizing weighted flow times and delivery costs", *Applied Mathematical Modelling*, Vol. 35, No. 1, (2011), 563-570.
- Zhong, X. and Jiang, D., "Integrated scheduling of production and distribution with release dates and capacitated deliveries", *Mathematical Problems in Engineering*, Vol. 2016, (2016).
- Sanei, M., Mahmoodirad, A., Niroomand, S., Jamalian, A. and Gelareh, S., "Step fixed-charge solid transportation problem: A lagrangian relaxation heuristic approach", *Computational and Applied Mathematics*, (2015), 1-21.
- Sahraeian, R. and Zabihi, F., "Trucks scheduling in a multi-product cross docking system with multiple temporary storages and multiple dock doors", *International Journal of Engineering-Transactions B: Applications*, Vol. 29, No. 11, (2016), 1595-1562.
- Guerrero, H.H. and Kern, G.M., "How to more effectively accept and refuse orders", *Production and Inventory Management*, Vol. 29, No. 4, (1988), 59-63.
- Slotnick, S.A., "Order acceptance and scheduling: A taxonomy and review", *European Journal of Operational Research*, Vol. 212, No. 1, (2011), 1-11.
- Nobibon, F.T. and Leus, R., "Exact algorithms for a generalization of the order acceptance and scheduling problem in a single-machine environment", *Computers & Operations Research*, Vol. 38, No. 1, (2011), 367-378.
- Og, C., Salman, F.S. and Yalçın, Z.B., "Order acceptance and scheduling decisions in make-to-order systems", *International Journal of Production Economics*, Vol. 125, No. 1, (2010), 200-211.
- Cesaret, B., Oğuz, C. and Salman, F.S., "A tabu search algorithm for order acceptance and scheduling", *Computers & Operations Research*, Vol. 39, No. 6, (2012), 1197-1205.
- Wang, X., Xie, X. and Cheng, T., "Order acceptance and scheduling in a two-machine flowshop", *International Journal of Production Economics*, Vol. 141, No. 1, (2013), 366-376.
- Esmailbeigi, R., Charkhgard, P. and Charkhgard, H., "Order acceptance and scheduling problems in two-machine flow shops: New mixed integer programming formulations", *European Journal of Operational Research*, (2015).
- Lin, S.-W. and Ying, K.-C., "Order acceptance and scheduling to maximize total net revenue in permutation flowshops with weighted tardiness", *Applied Soft Computing*, Vol. 30, (2015), 462-474.
- Rahman, H.F., Sarker, R. and Essam, D., "A real-time order acceptance and scheduling approach for permutation flow shop problems", *European Journal of Operational Research*, Vol. 247, No. 2, (2015), 488-503.
- Lei, D. and Guo, X., "A parallel neighborhood search for order acceptance and scheduling in flow shop environment", *International Journal of Production Economics*, Vol. 165, (2015), 12-18.
- Emami, S., Moslehi, G. and Sabbagh, M., "A benders decomposition approach for order acceptance and scheduling problem: A robust optimization approach", *Computational and Applied Mathematics*, (2016), 1-45.
- Wang, X., Huang, G., Hu, X. and Cheng, T.E., "Order acceptance and scheduling on two identical parallel machines",

- Journal of the Operational Research Society*, Vol. 66, No. 10, (2015), 1755-1767.
25. Thevenin, S., Zufferey, N. and Potvin, J.-Y., *A multi-coloring approach for an order acceptance and scheduling problem with preemption and job incompatibilities*. (2013), Technical report, CIRRELT--45.
 26. Noroozi, A., Mokhtari, H. and Abadi, I.N.K., "Research on computational intelligence algorithms with adaptive learning approach for scheduling problems with batch processing machines", *Neurocomputing*, Vol. 101, (2013), 190-203.
 27. Ghosh, J.B., "Job selection in a heavily loaded shop", *Computers & Operations Research*, Vol. 24, No. 2, (1997), 141-145.
 28. Herbots, J., Herroelen, W. and Leus, R., "Dynamic order acceptance and capacity planning on a single bottleneck resource", *Naval Research Logistics (NRL)*, Vol. 54, No. 8, (2007), 874-889.
 29. Park, J., Nguyen, S., Zhang, M. and Johnston, M., "Genetic programming for order acceptance and scheduling", in *Evolutionary Computation (CEC)*, IEEE Congress, (2013), 1005-1012.
 30. Xiao, Y.-Y., Zhang, R.-Q., Zhao, Q.-H. and Kaku, I., "Permutation flow shop scheduling with order acceptance and weighted tardiness", *Applied Mathematics and Computation*, Vol. 218, No. 15, (2012), 7911-7926.
 31. Qi, L., Jiang, D. and Wang, Z., "A modified discrete particle swarm optimization for a multi-level emergency supplies distribution network", (2016).
 32. A. Shahbahrani, S.J.a.S.H.S.Z., " Parallel implementation of particle swarm optimization variants using graphics processing unit platform", *International Journal of Engineering*, Vol. 30, No. 1, (2017), 48-56.
 33. Mehta, S.V. and Uzsoy, R., "Minimizing total tardiness on a batch processing machine with incompatible job families", *IIE Transactions*, Vol. 30, No. 2, (1998), 165-178.
 34. Morton, T. and Vepsalainen, A., "Priority rules and leadtime estimation for job shop scheduling with weighted tardiness costs", *Management Science*, Vol. 33, (1987), 1036-1047.
 35. M'Hallah, R., "Minimizing total earliness and tardiness on a single machine using a hybrid heuristic", *Computers & Operations Research*, Vol. 34, No. 10, (2007), 3126-3142.
 36. Pinedo, M., "Scheduling: Theory, algorithms, and systems", *Prentice Hall, USA*, (2002).
 37. Gordon, V., Proth, J.-M. and Chu, C., "A survey of the state-of-the-art of common due date assignment and scheduling research", *European Journal of Operational Research*, Vol. 139, No. 1, (2002), 1-25.

Coordinating Order Acceptance and Batch Delivery for an Integrated Supply Chain Scheduling

A. Noroozi^a, M. Mahdavi Mazdeh^a, M. Rasti-Barzoki^b

^a Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran

^b Department of Industrial and Systems Engineering, Isfahan University of Technology, Isfahan, Iran

P A P E R I N F O

چکیده

Paper history:

Received 12 January 2017

Received in revised form 17 February 2017

Accepted 10 March 2017

Keywords:

Order Acceptance

Integrated Production-distribution

Batch Delivery

Genetic Algorithm

Particle Swarm Optimization

در مقاله حاضر، در راستای توسعه مسائل یکپارچگی تولید و توزیع در زنجیره تأمین، هماهنگی پذیرش سفارشات و تحویل دسته‌ای را پیشنهاد داده است. هدف مقاله، هماهنگی: (۱) رد کردن تعدادی از سفارش‌ها، (۲) زمانبندی تولید سفارش‌ها پذیرفته شده و تحویل دسته‌ای آنها می‌باشد به طوریکه سود خالص کل ماکزیمم شود. برای مسئله در نظر گرفته شده، یک مدل ریاضی برنامه‌ریزی عدد صحیح ارائه شده است. علاوه بر این، یک الگوریتم فراابتکاری ترکیبی برای حل مسئله در اندازه‌های بزرگ توسعه داده شده است. در این الگوریتم، یک الگوریتم ابتکاری جهت دسته‌بندی، و جهت جستجوی بهتر همسایگی بهترین جواب هر نسل، یک الگوریتم جستجوی محلی پیشنهاد شده است. با توجه به اهمیت تأثیر جواب‌های اولیه در نتایج الگوریتم-های فراابتکاری، دو الگوریتم ابتکاری در تولید جمعیت اولیه پیشنهاد شده است. جهت بررسی صحت مدل پیشنهادی، مدل با استفاده از یک نرم‌افزار حل شده و جهت ارزیابی عملکرد الگوریتم فراابتکاری و تأثیر الگوریتم‌های پیشنهادی روی نتایج آن، داده‌های تصادفی تولید شده است.

doi: 10.5829/idosi.ije.2017.30.05b.10