The production-distribution problem with order acceptance and package delivery: models and algorithm

Majid Khalili^{1,*}, M. Esmailpour¹, and B. Naderi²

¹ Department of Industrial Engineering, Islamic Azad University of Karaj, Moazen Blvd, Shoora Street, Karaj, Iran

² Department of Industrial Engineering, Faculty of Engineering, Kharazmi University, Enghelab Street, South Mofateh Street, Tehran, Iran

Received 13 December 2015 / Accepted 8 October 2016

Abstract – The production planning and distribution are among the most important decisions in the supply chain. Classically, in this problem, it is assumed that all orders have to be produced and separately delivered; while, in practice, an order may be rejected if the cost that it brings to the supply chain exceeds its revenue. Moreover, orders can be delivered in a batch to reduce the related costs. This paper considers the production planning and distribution problem with order acceptance and package delivery to maximize the profit. At first, a new mathematical model based on mixed integer linear programming is developed. Using commercial optimization software, the model can optimally solve small or even medium sized instances. For large instances, a solution method, based on imperialist competitive algorithms, is also proposed. Using numerical experiments, the proposed model and algorithm are evaluated.

Key words: The production planning and distribution problem, Order acceptance, Scheduling, The package delivery, Imperialist competitive algorithm

1. The research's topic

Generally, the production planning is done in the short-term, mid-term and long-term of the three horizons of the time. The scheduling contains: the decision making in related to the production planning in the short-term time horizon which allocates the production capacity to the production targets. Its output is the exact determination of this issue that in which time and on what machine it should be process.

Deciding about the good's delivery to the customers that it is a process in the short-term time horizon, – along with the supply and the production issues – is of the important decisions of every production complex. The package delivery is one of the delivery approaches. The delivery process is always done next to the production process. On the other hand, in the cases of we are required to respecting the delivery time determined by the customer, we obligate to coordinating the production program with the delivery requirements. As a result, the production and the delivery decisions are totally related to each other, so it is better that these two issues would be checked as integrated and simultaneous. In considering of the scheduling's issue in the production and the package delivery approach to the delivery subject, the integrated issue will be contain the scheduling simultaneous determination and the package

delivery. In the short-term time horizon, the available capacity of the production system is limited and usually is less than the necessary capacity for processing of all of the orders received by the system. So, a correct decision-making for the production planning, in addition to the scheduling determination and the works' implementation sequence, includes the choice of the best orders from the orders' basket received by the system. The approach of the orders' choice from the received orders' basket, is named so-called the order's acceptance. The main idea of this study is presenting of an integrated issue in order to the simultaneous determination of the accepted orders for producing, the production sequence determination and the orders delivery to the customers with the package delivery approach.

2. The issue definition

A production system which works with the production strategy based on the order, in a planning period with having the short-term time horizon of (T), receives several orders (i) from several customers. These orders (O_{ij}) have the processing time of (P_i), revenue (R_i) and the production cost (PC_i). It assumes that the available time is not enough for all of the received orders. So, in attention to the time's limitation, the producer likes accept the sales revenue and the production

*e-mail: khalili.mj@gmail.com

system's costs of the best orders and plans for their processing and then delivery the orders to the customers. Delivering the orders to the customers that should be happen, in maximum, until the end of the period, is done simultaneously with the package delivery approach.

Each package can only contain the orders belong to one customer. The producer, after the processing completion of each order, will have three possibilities: at first, to send the order for the customer, immediately. In this case, we will have a package containing an order. Second, to send the processed order to the storehouse for processing the other orders that belong to the customer of the same order and then all of them would be sent in the form of one package. Third, the order which was processed, recently along with the orders that have already processed (and those belong to one customer) would be sent in form of a package. In this case, the order that has been processed recently, do not go to the storehouse, longer. If an order would be sent to the storehouse after processing, its handling cost (HC_i) should be paid. This handling cost would be assumed depend on the order accumulation time. Totally, four categories of the cost are detected in this production system: the production's cost, the preparation cost depend on the sequence, the handling cost and the cost of delivering the package which is depend on the number of the sent packages and independent of the number of the orders of inside the package. The objective function is the issue of maximizing the profit.

This profit achieves from the subtraction of the sales revenue and the costs.

3. Literature review

The first research into the issue of scheduling and acceptance of the order was performed by Aslatnyk and Morton (1996) and Qsht (1997). In the study of Aslatnyk and Morton the manufacturing environment is single-machine and all orders are present at the zero time in the production system [1]. Lewis and Slotnick investigated the multi-period scheduling and acceptance of the order. In this research, the planning horizon is divided into a number of courses [2]. Moghaddam et al. recruited eight algorithms to solve the problem of single-machine scheduling and order rejection of that three algorithms are based on developing the simulated annealing method and five of them are according to the non-dominated sorting genetic algorithm (NSGA) one [3]. Mokhtari developed a meta-intelligent Algorithm of water droplets for solving his bi-objective problem in the environment of parallel machines [4]. Wang et al. investigated the problem of timing and acceptance of the order with the possibility of outsourcing in a few modes. The objective function is equal to accepted order minus the cost of outsourcing [5]. Lu et al. presented the accurate polynomial time algorithm for optimal solution of scheduling problem and of rejecting the order in a single machine mode [6]. Zhong and colleagues looked over the scheduling problem and order rejection in single-machine mode assuming the absence of a machine in some period of time [7]. Moghaddam and colleagues examined the timing and order rejection in the form of a bi-objective optimization problem [3]. He and his colleagues studied the order and timing in the environment

of parallel machines. This order is defined for the " m " number of parallel machine and the " n " number of orders [8]. Mokhtari studied the issue of minimizing the number of delayed orders (accepted orders) and fine rejects (for orders rejected) in the environment of parallel machines [4].

Khalili and Moghadam examined a scheduling problem in a "flow shop" environment that the objective function is the issues of minimizing "makespan" and "Total Weighted Tardiness". In this issue all work may not be processing by all the available machines. Hereupon, they used the simulated annealing algorithm to solve the problem [9]. Khalili examined a scheduling issue in a "hybrid no-wait flowshop" environment that the objective function is the issue of minimizing "makespan" and "Total Tardiness" [10]. The timing issue and package delivery was determined first in 1993 by Kalbachr and Chang. He is the analysis of a single machine with the objective function minimizing total weighted earliness and delivery costs were closed [11]. The main subject of their analysis was the issue of single machine problem with the objective function of minimizing the total weighted earliness and also the package delivery cost [11]. Wang and Cheng also examined the issue of timing and package delivery on parallel machines in 2000 [12]. Mahdavi Mazdeh and colleagues applied the objective function of minimizing the total current time work and delivery costs for scheduling tasks in the single machine mode [13]. Ji et al. showed that scheduling problem and package delivery on the single machine and with the objective function of minimizing the total cost of current work load and delivery costs is a NP-hard strongly problem [14]. Mahdavi Mazdeh and colleagues investigated the issue of scheduling and package delivery on the single machine with the assumption that works are not necessarily available at the zero time and each work consists the time of arrival to the workshop [15]. Mahdavi Mazdeh and his colleagues studied the single-machine scheduling and package delivery with the objective function of total weighted current work time order and package delivery cost [16]. Hamidinia and colleagues to solve the problem of single-machine scheduling and package delivery proposed a non-linear model using genetic algorithms [17]. Selvarajah and Zhang developed an innovative solution for scheduling and package delivery problem with the objective function of total weighted current work time order and package delivery cost [18].

4. Design models and solving methods

4.1. Problem assumptions

1. In a short period of time T , the manufacturer intends to submit orders among incoming orders that considering time constraints, system costs, and earnings per order, the most profits would be gained.
2. The manufacturing environment is single-machine and it is continuously available during the planning period.
3. At any time, only one order is processed.
4. Interruption of work (work breakdown) is not allowed.
5. Before any order processing, preparation is necessary that its duration is dependent on the sequence.
6. The manufacturer is obliged to deliver customer orders up to the end of the period.

7. Delivering the package is performed according to the package delivery approach.
8. Packages have no capacity constraints.
9. The costs of the system are: the cost of production (processing), maintenance costs and preparation costs (considered to be a multiple of preparation time), delivery costs (depending on the number of packages).
10. The objective function is to maximize the benefits derived from the difference between sales revenue minus the cost of the entire system.

4.2. Mathematical model

In this study, design of simultaneously scheduling model, the acceptance of orders and deliveries, considering the capacity constraints, sequence-dependent setup and package delivery approaches are discussed. This model is based on the Mixed Integer Linear Programming (MILP). For describing the above model, the index, parameters and variables used are shown in Table 1.

The mathematical model is presented below:

$$\begin{aligned} \text{Max } Z = & \sum_i (R_i - PC_i)x_i - \sum_i \sum_{i'} SC_{ii'} y_{ii'} \\ & - \sum_i (del_i - c_i) HC_i - \sum_j DC_j \alpha_j \end{aligned} \quad (1)$$

Subject to

$$c_i \leq Tx_i \quad \forall i \quad (2)$$

$$c_i = s_i + P_i \quad \forall i \quad (3)$$

$$s_{i'} \geq c_i + ST_{ii'} y_{ii'} - M(1 - y_{ii'}) \quad \forall i, \forall i' \quad (4)$$

$$\sum_i \sum_{i'} y_{ii'} = \sum_i x_i - 1 \quad \forall i, \forall i' \quad (5)$$

$$\sum_{i'} y_{i'i} \leq x_i \quad \forall i, \forall i' \quad (6)$$

$$\sum_i y_{ii'} \leq x_i \quad \forall i, \forall i' \quad (7)$$

$$\sum_k z_{ik} = x_i \quad \forall i \quad (8)$$

$$\sum_j z_{jk} = 1 \quad \forall k \quad (9)$$

$$\sum_i z_{ik} O_{ij} \leq Mz_{jk} \quad \forall j, \forall k \quad (10)$$

Table 1. Parameters and variables of decision.

Index/parameters/ variables	Descriptions
i	Order index
j	Customer value
k	Batch index
T	Planning horizon
PC_i	Order i Production cost
$SC_{ii'}$	Order i' Preparation cost if order i is before that
HC_i	Order i Maintenance cost
R_i	Order i income
P_i	Order i processing time
DC_j	Each time delivery cost to the customer j
O_{ij}	Equal 1: if order i is related to customer j otherwise equal 0
$ST_{ii'}$	Order i' Preparation time if order i is before that
S_i	Order i start time
C_i	Order i completion time
del_i	Order i delivery time
C_k	Batch k completion time
A_j	Quantity of delivered batch to the customers
X_i	Equal 1: if order i is accepted otherwise equal 0
Z_{ik}	Equal 1: if order i is related to batch k otherwise equal 0
Z_{jk}	Equal 1: if batch k is related to customer j otherwise equal 0
$Y_{ii'}$	Equal 1: if order i' is processed after order i otherwise equal 0

$$\sum_i z_{ik} O_{ij} > -Mz_{jk} \quad \forall j, \forall k \quad (11)$$

$$c_k \geq c_i - M(1 - z_{ik}) \quad \forall i, \forall k \quad (12)$$

$$del_i \geq c_k - M(1 - z_{ik}) \quad \forall i, \forall k \quad (13)$$

$$\alpha_j = \sum_k z_{jk} \quad \forall k \quad (14)$$

$$\begin{aligned} c_i, s_i, c_k &\geq 0 \\ x_i, z_{ik}, z_{jk}, y_{ii'} &= 0, 1 \quad \forall i, \forall j, \forall k, \forall i' \\ \alpha_j &= \text{Integer} \end{aligned} \quad (15)$$

The equation (1) shows the objective function maximize profit. The profit is obtained from the difference between sales revenue minus the cost (maintenance, preparation, production and delivery). The equation (2) shows that if i order is accepted, the order shall be terminated before the end of the (T) time horizon and if the i order is not accepted the completion time order does not mean and is considered as zero. The equation (3) shows that if i order is accepted, the complete time order of i equal to start time plus processing time and if i order is not be accepted, start time is equal to end time.

The equation (4) shows that if the i' order to be processed after the order i , the i' start order at least is equal to the time of completion the task i plus the i' preparation time work, and if the i' order is not processing after i order, the restrictions will be thwarted. The equation (5) suggests that the number of times that the preparation is done, is equal to the number of orders accepted, except to the first work in which the preparatory work is not done. The equation (6) shows if i order to be accepted, certainly is considered somewhere in the production sequence and if not, there is no place in the production sequence for it. The equation (7) shows if i order to be accepted, certainly is considered somewhere in the production sequence and if not, there is no place in the production sequence for it. The equation (8) shows that if i order to be accepted it must be assigned to a batch like k and if i order is not accepted it has no place in any of the batches delivered. The equation (9) indicates that each batch such as k can only include customer orders like costumer j . The equations (10) and (11) requiring the model that only if k batch belongs to the customer j , the customer orders like i belong to customer j can be allocated to k batch and contrary, if the customer orders like i belong to customer j be placed in k batch then k batch will be allocated to customer j . The equation (12) indicates the completion of k batch, is at least equal to the time of completion of the last job within the batch (if k batch includes i order). The equation (13) shows that the delivery time of i , is at least equal to the completion of the k batch (if k batch includes i order). The equation (14) represents the total number of batches delivered to the customer j . The equation (15) shows the variable types.

4.3. Development of solving method

4.3.1. Exact method

Due to the linearity of the mathematical model, by using the Gomez software, the optimal solution can be obtained. Considering that for the large size it is very time-consuming that problem to be solved, in this way solving time is limited to 3600 s. According to the state to solve every problem with precision, the issues are divided into three categories: small issues (problems to optimally be solved by exact method in 3600 s), medium (problems that are unlikely to be solved in 3600 s by exact method, but at least a feasible solution is found for them) and large (problems that no feasible solution cannot be found for them over 3600 s).

4.3.2. Imperialist competitive algorithm

Imperialist is the policy of extending the power and rule of a government beyond its own boundaries. A country may attempt to dominate others by direct rule or less obvious means such as a control of markets for goods or raw materials. The latter is often called Neocolonialism. In its initial forms imperialism was just a political control over other countries in order to only use their sources. Also in some cases the reason to control another country was just preventing the opponent imperialist from taking possession of it. No matter what the reason was the imperialist states were competing

strongly for increasing their colonies and spreading their empires over the world. This competition resulted in a development of powerful empires and the collapse of weaker ones [19].

Figure 1 shows the flowchart of the proposed algorithm [19].

Generating initial empires

The goal of optimization is to find an optimal solution in terms of the variables of the problem. We form an array of variable values to be optimized. This array is called "country". In an N_{var} dimensional optimization problem a country is a $1 \times N_{\text{var}}$ array. This array is defined by

$$\text{Country} = [p_1, p_2, \dots, p_{N_{\text{var}}}] \quad (16)$$

The variable values in the country are represented as floating point numbers. The cost of a country is found by evaluating the cost function f at the variables $(p_1, p_2, \dots, p_{N_{\text{var}}})$ then

$$\text{Cost}_i = f(\text{Country}_i) = f(p_1, p_2, \dots, p_{N_{\text{var}}}) \quad (17)$$

To start the optimization the algorithm we generate the initial population of size N_{pop} . We select N_{imp} of the most powerful countries to form the empires. The remaining of N_{col} of the population will be the colonies each of which belongs to an empire. Then we have to types of countries: imperialist and colony. To form the initial empires, we divided the colonies among imperialists based on their power. That is the initial number of colonies of an empires should be directly proportionate to its power. To divide the colonies among imperialists proportionally, we define the normalized cost of an imperialist by

$$C_n = \max \{C_i\} - c_n \quad (18)$$

where c_n is the cost of n th imperialist C_n is its normalized cost. Having the normalized cost of all imperialists, the normalized power of each imperialist is defined by

$$P_n = \frac{|C_n|}{\sum_{i=1}^{N_{\text{imp}}} C_i} \quad (19)$$

From another point of view, the normalized power an imperialist is the portion of colonies that should be possessed by that imperialist. Then the initial number of colonies of an empire will be

$$N.C._n = \text{round}\{p_n \cdot (N_{\text{col}})\} \quad (20)$$

where $N.C._n$ is the initial number of colonies of n th empire and N_{col} is the number of all colonies. To divide the colonies, for each imperialist we randomly choose $N.C._n$ of the colonies and give them to it. Figure 2 shows how initial empires are Generating.

Moving the colonies of an empires of an empire toward the imperialist

Imperialist countries started to improve their colonies. We have modeled this fact by moving all the colonies toward the imperialist. This movement is shown in Figure 3, in which the colony moves toward the imperialist by x units. The new

Figure 1. Flowchart of the proposed algorithm.

Figure 2. Generating initial empires.

position of colony is shown in a darker color. The direction of the moment is the vector from colony to imperialist. In this figure x is a random variable with uniform distribution. Then for x we have

$$x \sim U(0, \beta \times d) \quad (21)$$

where β is a number greater than 1 and d is the distance between colony and imperialist. A $\beta > 1$ causes the colonies to get closer to the imperialist state from both sides.

To search different points around the imperialist we add a random amount of deviation to the direction of movement. Figure 4 shows the new direction. In this figure θ is a random number of with uniform (or any proper) distribution. Then

$$\theta \sim U(-\gamma, \gamma) \quad (22)$$

where γ is a parameter that adjust the deviation from the original direction. Nevertheless the values of β and γ are arbitrary, in most of our implementation a value of about 2 for β and about $\pi/4$ for γ have resulted in good convergence of counties to the global minimum.

Exchanging positions of the imperialist and a colony

While moving toward the imperialist, a colony may reach to a position with lower cost than that of imperialist. In such a case the imperialist moves to the position of that colony and vice versa. Then algorithm will continue by the imperialist in a new position and then colonies start moving toward this position. Figure 5 shows how a colony exchange the positions of the imperialist.

Total power of an empire

Total power of an empire is mainly effected by the power of imperialist country. But the power of the colonies of an empire has an effect on the total power of that empire. We have modeled this fact by defining the total cost by

$$T.N.C._n = \text{Cost}(\text{imperialist}_n) + \xi \text{ mean}\{\text{Cost}(\text{Colonies of impire}_n)\} \quad (23)$$

where $T.N.C._n$ is the total cost of the n th empire and ξ is a positive number which is considered to be less than 1. A little value for ξ causes the total power of the empire to be determined by just the imperialist and increase the rule of the colonies in determining the total power of an empire. We have used the value of 0.1 for ξ in most of our implementation.

Imperialistic competition

All empires try to take possession of colonies of other empires and control them. This imperialistic competition gradually brings about a decrease in the power of weaker empires and an increase in the power of more powerful ones. We model this competition by just picking some (usually one) of the weakest colonies of the weakest empires and making a competition among all empires to possess these (this) colonies. To start the competition, first, we find the possession probability of each empire based on its total power. The normalized total cost is simply obtained by

$$N.T.C._n = \max\{T.C._i\} - T.C._n \quad (24)$$

where $T.C._n$ and $N.T.C._n$ are respectively total cost and normalized total cost of n th empire. Having the normalized total cost, the possession probability of each empire is given by

$$P_{p_n} = \frac{|N.T.C._n|}{\sum_{i=1}^{N_{\text{imp}}} N.T.C._i} \quad (25)$$

Figure 3. Moving colonies toward their relevant imperialists.

Figure 5. Exchanging positions of the imperialist and a colon.

Figure 4. Moving colonies toward their relevant imperialists in a randomly deviated direction.

Figure 6. Eliminating the powerless empires.

To divide the mentioned colonies among empires based on the possession probability of them, we form the vector P as

$$P = (p_{p_1}, p_{p_2}, \dots, p_{p_{N_{\text{imp}}}}) \quad (26)$$

Then we create a vector with the same size as P whose elements are uniformly distributed random numbers.

$$R = (r_1, r_2, \dots, r_{N_{\text{imp}}}) \quad (27)$$

$$r_1, r_2, \dots, r_{N_{\text{imp}}} \sim U(0, 1) \quad (28)$$

Then we form vector D by simply subtracting R from P .

$$\begin{aligned} D &= P - R = (D_1, D_2, \dots, D_{N_{\text{imp}}}) \\ &= (p_{p_1} - r_1, p_{p_2} - r_2, \dots, p_{p_{N_{\text{imp}}}} - r_{N_{\text{imp}}}) \end{aligned} \quad (29)$$

Referring to vector D , we will hand the mentioned colonies to an empire whose relevant index in D is maximum.

Eliminating the powerless empires

Powerless empires will collapse in the imperialistic competition and their colonies will be divided among other empires. In modeling collapse mechanism different criteria can be defined for considering an empire powerless. In most of our implementation, we assume an empire collapsed and eliminated it when it loses all of its colonies. Figure 6 shows how a powerless empire is eliminated.

Convergence

After a while all the empires except the most powerful one will collapse and all the colonies will be under the control of this unique empire. In this ideal new world all the colonies will have the same positions and same costs and they will be controlled by an imperialist with the same position and cost as themselves. In this ideal world, there is no difference not only among colonies but also between colonies and imperialist. In such a condition we put an end to the imperialistic competition and stop the algorithm. In the next part we apply the proposed algorithm to some of benchmark problems in the optimization. The main steps in the algorithm are summarized in the pseudo-code shown in Figure 7.

- 1) Select some random points on the function and initialized the empires.
- 2) Move the colonies toward their relevant imperialist. (Assimilating)
- 3) If there is a colony in an empire which has lower cost than that of imperialist, exchange the positions of that colony and the imperialist.
- 4) Compute the total cost of all empires.(Related to the power of both imperialist and its colonies)
- 5) Pick the weakest colony (colonies) from the weakest empire and give it (them) to the empire that has the most likelihood possess it.(Imperialistic competition)
- 6) Eliminate the powerless empire.

Figure 7. Pseudo-code for the proposed algorithm.

4.4. Application example

In this section a numerical example in small size of model is shown. Table 2 gives information about demand, costs and revenue of each order. Table 3 gives information about set up time for orders.

($T = 20$) & ($SC = 5$ ST).

To solve the problem, at first we should accept the orders based on indices T , P_i , ST :

Order	1	2	3	4	5
$(R_i - PC_i)$	80	35	60	50	50
Sequencing	1	3	4	5	2
P_i	3	4	5	3	6

Because of $T < 20$, order 2 should be eliminated (Figures 8 and 9). Therefore sequencing convert to:

Sequencing	1	3	4	5
ST	1	4	5	3

Now, suppose orders are delivered by these batches:

- Order 1 to customer 1.
- Orders 4 & 5 to customer 2.
- Order 3 to customer 3.

Order	1	4	5	3	Total
$(R_i - PC_i)$	80	60	50	50	240
SC_i		2×5			10
DC_j		$(2 \times 25) + (1 \times 20)$			70
$(del_i - c_i)HC_i$		1.5×3.5			5.25

$$Z = 240 - 10 - 70 - 5.25 = 154.75.$$

4.5. Numerical results and analysis

The 23 randomly generated samples are distinct based on two parameters: the number of orders (i) and customer number (j).

4.5.1. Exact method

In this method, the mathematical model is designed to determine the order at the reception, scheduling and delivery of a package with the aim of finding the optimal solution in a Gomez software environment. Considering that solving problem for large size is very time consuming, in this way it is limited to 3600 s. Table 3 shows the results of a detailed method of problem solving.

4.5.2. Colonial competition methods

Meta heuristic correct functioning depends on set parameters. Therefore, before using this colonial competition method first it is necessary to set the parameter

First, for iteration parameter a value of 100 and for β coefficient value of 0.8 is considered, then for each parameter number of value is selected based on trial and error method. Table 2 shows the values of various parameters. Based on Table 4 a total of 12 different combinations for algorithm parameters that a compound should be selected among them. To choose the best combination of parameters, the number of five issues of the optimal sample issues are available using the independent component analysis (ICA) algorithm, with 12 different parameter combinations been solved and the solution obtained been compared with the optimal solution. Any combination of parameters which provides closest solution to the optimal solutions in solving these five problems is selected. ICA algorithm similarity with the optimal solution is measured using relative deviation Total Relative Gap (TRG) indicators.

$$TRG_q = \sum_{p=1}^5 \frac{\text{opt}_p - S_{pq}}{\text{opt}_p} \times 100 \quad (30)$$

TRG_q values for 12 combinations of parameter shown in Table 4, the lowest amount is for the compound (5). Hereupon, this algorithm parameter values is chosen according to below: $nPop = 20$, $nEmp = 7$, $kesi = 1/5$.

5. Comparison and result analysis

5.1. Comparison in the case of the optimal solution

Based on the observations, for small size problems, the optimal solution using the exact method is achieved in less

Table 2. Demand, costs and revenue of each order.

Index/parameter/variable	Description				
j	Customer 2		Customer 1		
O_{ij}	O52	O42	O31	O21	O11
P_i	3	5	4	6	3
R_i	75	80	70	50	100
PC_i	25	30	10	15	20
PC_i	2	1	3	1.5	2
DC_j	20		25

Figure 8. Sequencing of orders.

time than one hour. Table 5 provides a comparison between the answers obtained from two solutions and shows the relative deviation from the optimal solution for ICA method. The relative deviation according to equation (31) is obtained.

$$\text{Relative deviation} = \frac{\text{opt} - \text{sol}}{\text{opt}} \times 100 \quad (31)$$

In this equation opt is optimum value according to these exact method and sol is observed value based on the ICA method.

5.2. Comparison in the absence of optimal solution

Since the optimal solution is not available in medium and large size problems, the comparison of observed results are discussed. Table 6 shows the objective function for two solution method. In the ICA method, the best objective function value observed for each sample is given.

$$\text{Relative deviation} = \frac{\text{ICA best} - \text{sol}}{\text{ICA best}} \times 100 \quad (32)$$

The relative deviation of the objective function value according to equation (32) calculated and presented.

In this equation, the ICA best is the function objective value according to the ICA method and sol is the objective function value obtained according the exact method.

5.3. Analysis of the results

5.3.1. Evaluation methods of solutions

For the problems of small size, the optimal solution is obtained using the exact method in limited time to 3600 s.

Table 3. Set up times for orders.

	1	2	3	4	5
1	0	1	1	0.5	1
2	0.5	0	1.5	1	1
3	1	1	0	0.5	1.5
4	0.5	0.5	1	0	1.5
5	1	0.5	1	0.5	0

Figure 9. Sequencing of orders.

The best response in all matters of ICA method is equal to the optimal solution. Therefore, for the small size problems, we recommend using both exact solution and ICA. In medium and large size problems with the enlargement of the scale of the problem, gradually ICA method is to produce better solutions than the exact method.

The reason for this is that, in line with the enlargement of the problem scale and the exponential increase in time required for the optimal solution, the solution technique which is limited to 3600 s, compared to the time required for the optimal solution would be negligible and this fact improve the quality of the results with the ICA method.

5.3.2. Benefits of batch delivery

In literature, two criteria are common: (1) delivery after production and (2) delivery at the end of period.

But, batch delivery, according to the definition, is a more comprehensive approach compared to the previous approach. In batch delivery maybe a product to be sent for a customer with a batch. This is equivalent with delivery after production. By the way, it maybe all goods to be sent for a customer with a package. This is equivalent with delivery at the end of period. Despite comprehensive batch delivery, to verify this way than two others, a comparison between three methods is done.

Numerical example

In Section 4.4 a numerical example is investigated. Comparison between revenue, production cost, holding cost, delivery cost, set-up cost and benefit in three methods is shown in Table 7.

Small size problems

The approach used in comparison between three methods in numerical example is applied for small size problems. Table 8 is shown results of three methods.

Table 4. Comparison between different combinations of parameters of ICA method.

	1	2	3	4	5	6	7	8	9	10	11	12
nPop	20	20	20	20	20	20	50	50	50	50	50	50
nEmp	3	3	5	5	7	7	3	3	5	5	7	7
kesi	1.5	1.8	1.5	1.8	1.5	1.8	1.5	1.8	1.5	1.8	1.5	1.8
Pro. $i-j$												
4-2	95	100	97.5	100	95	100	102	92	90	105	95	100
5-2	85	100	120	117.5	120	100	155	155	115	130	135	115
7-3	145	155	170	190	170.5	186	170.5	195	190	210	200	180
8-3	195	210	190	190	217.5	230	215	220	230	230	235	220
9-4	240	250	247	252	247	255	265	260	240	242	250	247
TRG _q	0.16	0.09	0.84	0.16	0	0.47	1.34	0.48	0.45	0.15	0.26	0.67

Table 5. Comparison of the results obtained from two methods for solving the problems of small size.

Order index	Customer index	Results in precise method	Results in ICA method	Relative deviation
4	2	95	95	0
5	2	120	120	0
7	3	170.5	170.5	0
8	3	217.5	217.5	0
9	4	247	247	0
10	4	274	274	0
12	4	409	409	0

Table 6. Comparison of two solution methods for solving problems of medium-sized and large-sized samples.

Order index	Customer index	Results in precise method	Results in ICA method	Relative deviation
17	5	523.5	495	—
23	6	631.75	648.5	2
26	7	724.5	733	1
31	8	846	872.5	2
37	9	952	891.5	—
44	10	1225	1280	4
48	11	1196.75	1269	5
55	12	1373	1410	2
61	13	1527	1608	5
69	14	1050.5	1435	20
77	15	1411.5	1542	8
82	16	1477	1721	10
87	17	625.5	1911	38
93	18	1192	1788	30
98	19	2304	2310	1
100	20	781	1988.5	48

Table 7. Comparison between benefits of three methods in numerical example.

	Results in batch delivery	Results in delivery after production	Results in delivery at the end of period
$(R_i - PC_i)$	240	240	240
SC_i	10	10	10
$(del_i - c_i)HC_i$	5.25	0	42
DC_j	70	90	40
Z	154.75	140	148

Table 8. Comparison between benefits of three methods in small size problems.

Order index	Customer index	Results in batch delivery	Results in delivery after production	Results in delivery at the end of period
4	2	95	88	90
5	2	120	111	115
7	3	170.5	160	166
8	3	217.5	206	210.5
9	4	247	232	241
10	4	274	266	269
12	4	409	398	402

According to the results, batch delivery, always maximize benefit of a production system.

References

1. J.B. Ghosh, Job selection in a heavily loaded shop, *Computers & Operations Research* 24 (1997) 141–145.
2. H. Lewis, S. Slotnick, Multi-period job selection: planning workloads to maximize profit, *Computers & Operations Research* 29 (2002) 1081–1098.
3. A. Moghaddam, F. Yalaoui, L. Amodeo, Efficient meta-heuristics based on various dominance criteria for a single-machine bi-criteria scheduling problem with rejection, *Journal of Manufacturing Systems* 34 (2015) 12–22.
4. H. Mokhtari, A nature inspired intelligent water drops evolutionary algorithm for parallel processor scheduling with rejection, *Applied Soft Computing* 26 (2015) 166–179.
5. X. Wang, Q. Zhu, T. Cheng, Subcontracting price schemes for order acceptance and scheduling, *Omega* 54 (2015) 1–10, DOI: [10.1016/j.omega.2015.01.005](https://doi.org/10.1016/j.omega.2015.01.005).
6. L. Lu, C. Ng, L. Zhang, Optimal algorithms for single-machinescheduling with rejection to minimize the makespan, *International Journal of Production Economics* 130 (2011) 153–158.
7. X. Zhong, J. Ou, G. Wang, Order acceptance and scheduling with machine availability constraints, *European Journal of Operational Research* 232 (2014) 435–441.
8. J. Ou, X. Zhong, G. Wang, An improved heuristic for parallel machine scheduling with rejection, *European Journal of Operational Research* 241 (2015) 653–661.

9. M. Khalili, R. Moghaddam, A multi-objective electromagnetism algorithm for a bi-objective flowshop scheduling problem, *Journal of Manufacturing Systems* 31 (2012) 232–239.
10. M. Khalili, A multi-objective electromagnetism algorithm for a bi-objective hybrid no-wait flowshop scheduling problem, *The International Journal of Advanced Manufacturing Technology* 70 (2014) 1591–1601.
11. M. Mahdavi Mazdeh, A.N. Esfahani, S. Sakkaki, A.E. Pilerood, Single machine batch scheduling minimizing weighted flow times and delivery costs with job release times, *International Journal of Industrial Engineering Computations* 3 (2012) 347–364.
12. G. Wang, T. Cheng, Parallel machine scheduling with batch delivery costs, *International Journal of Production Economics* 68 (2000) 177–183.
13. M. Mahdavi Mazdeh, M. Sarhadi, K. Hindi, A branch-and-bound algorithm for single-machine scheduling with batch delivery minimizing flow times and delivery costs, *European Journal of Operational Research* 183 (2007) 74–86.
14. M. Ji, Y. He, T. Cheng, Batch delivery scheduling with batch delivery cost on a single machine, *European Journal of Operational Research* 176 (2007) 745–755.
15. M. Mahdavi Mazdeh, M. Sarhadi, K. Hindi, A branch-and-bound algorithm for single-machine scheduling with batch delivery and job release times, *Comput. Oper. Res.* 35 (2008) 1099–1111.
16. M. Mahdavi Mazdeh, S. Shashaani, A. Ashouri, K. Hindi, Single-machine batch scheduling minimizing weighted flow times and delivery costs, *Applied Mathematical Modelling* 35 (2011) 563–570.
17. A. Hamidinia, S. Khakabimamaghani, M. Mahdavi Mazdeh, M. Jafari, A genetic algorithm for minimizing total tardiness/earliness of weighted jobs in a batched delivery system, *Computers & Industrial Engineering* 62 (2012) 29–38.
18. E. Selvarajah, R. Zhang, Supply chain scheduling at the manufacturer to minimize inventory holding and delivery costs, *International Journal of Production Economics* 147A (2014) 117–124.
19. E. Atashpaz-Gargari, C. Lucas, Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition, *IEEE Congress on Evolutionary Computation (CEC 2007)* (2007) 4661–4667.

Cite this article as: Khalili M, Esmailpour M & Naderi B: The production-distribution problem with order acceptance and package delivery: models and algorithm. *Manufacturing Rev.* 2016, 3, 18.