

## An integrated model of scheduling, batch delivery and supplier selection in a make-to-order manufacturing system

Mohammad Mahdavi Mazdeh, Mehdi Heydari and Ayatollah Karamouzian\*

*Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran*

### CHRONICLE

#### *Article history:*

Received June 25, 2015  
 Received in revised format:  
 October 12, 2015  
 Accepted December 15, 2015  
 Available online  
 December 17 2015

#### *Keywords:*

*Scheduling*  
*Single-machine*  
*Suppliers selection*  
*Batch delivery*  
*Greedy heuristic*

### ABSTRACT

This paper analyzes a supply chain, which consists of a manufacturer, a retailer and several suppliers in which the retailer orders jobs to the manufacturer and the suppliers provide the requiring parts. The manufacturer schedules and processes the orders and dispatches them to the retailer either individually or collectively in batches. The manufacturer incurs a penalty cost for each tardy job and a transportation cost for every delivered batch and therefore, searches for a schedule that yields minimum number of tardy jobs and batches. Moreover, the manufacturer tries to optimize its supplying cost through locating the suppliers that offer appropriate release times and costs for manufacturing parts. Since the release times of parts directly affect scheduling of orders, in this research, we develop an integrated mathematical model for the manufacturer that incorporates suppliers' selection issue into the scheduling and batching decisions. Furthermore, we present a heuristic algorithm (greedy algorithm) and also a local search to quickly determine the optimal or near-optimal solutions. The computational analysis shows the importance of the integrated model and also the superiority and effectiveness of the heuristic algorithms.

© 2016 Growing Science Ltd. All rights reserved.

## 1. Introduction

In a make to order production environment, customers or retailers order different items to the manufacturers and expect to receive them on due dates. Generally, if a customer order is delivered before its due date, it is regarded as an on time job, otherwise it is a tardy job. Naturally, any tardiness of orders leads to customer dissatisfaction and therefore, the manufacturers try to schedule jobs so that no jobs become tardy. This problem has been known as a single machine scheduling problem with the objective of minimizing the number of tardy jobs which has been widely discussed in the past decades (Detienne, 2014). Nevertheless, there are usually many constraints and costs such as limited production capacity and transportation costs that prevent manufacturers to deal with the problem of scheduling of jobs, independently. The limited production capacity enforces the manufacturers to schedule jobs based on some criteria such as minimization of the number of tardy jobs while the high transportation costs

\* Corresponding author. Tel: +98-21-73225002; Fax: +98-21-73225098  
 E-mail address: [A\\_karamouzian@ind.iut.ac.ir](mailto:A_karamouzian@ind.iut.ac.ir) (A. Karamouzian)

lead them to deliver jobs in batches. Batch delivery reduces the transportation costs due to the decrease in the number of transshipment. However, it might cause tardiness of jobs. Hence, production scheduling, delivery or distribution of jobs are conflicting objectives. Traditionally, production decisions are first set in a firm and then distribution schedule is planned with little or no integration (Chen, 2010). Evidently, this procedure does not guarantee the best decisions and it needs to simultaneously consider the production schedule and delivery plan. Chen (2010), based on a survey on the relevant literature, states that although there is a huge amount of research on integrated production and outbound distribution models at the strategic and tactical levels, the research at the detailed scheduling where revenues, costs and customer service level are analyzed at the order level is fairly recent. It implies the need for conducting more studies into the subject.

In addition to necessity of integration of outbound decisions into internal scheduling, the issue of supplier selection needs to be taken into account since it has a major impact on the manufacturer's costs and also scheduling of orders. Generally, the manufacturers collaborate with several suppliers to provide the requiring parts. Usually, a job or a customer order cannot be processed until the corresponding parts are provided and thus the manufacturers expect to be supplied according to their production plans. However, in the real world, suppliers have their own constraints and might not be able to provide parts for manufacturers in the demanded time. They usually suggest different offers based on their production capacity and costs and therefore the manufacturers face with the problem of finding the most appropriate suppliers. Naturally, every supplier has a pros and cons. In the context of scheduling, a supplier can provide an order for a manufacturer sooner while the offering price might be higher. Moreover, due to different locations of suppliers, the collection costs might not be the same for the manufacturer when the suppliers deliver parts "Ex-works" (a trade term requiring the seller to deliver goods at his or her own place of business and all transportation costs are by the buyer). Consequently, the supplier can affect the manufacturer at least in two ways: First, the ready time for processing a job in the facility of the manufacturer is defined by the supplier. Second, any changes in the selection of suppliers may reduce or increase of collection costs for the manufacturer. Other factors such as quality of parts and purchasing costs can also be affected by altering suppliers.

In sum, it is concluded that for scheduling of jobs, outbound decisions consisting of selection of suppliers and distribution of completed orders cannot be regarded as independent tasks and should be performed together. In this research, we develop a single machine scheduling model with the objective of minimizing total number of tardy jobs and transportation costs and extend the scope of study from shop floor problem to the supply chain viewpoint. The contribution of the paper lies in considering machine scheduling, batch delivery to customers (retailers) and supplier selection (sourcing) in an integrated manner which is, to the best of authors' knowledge, is not discussed in the relevant literature. We present a mathematical formulation for this general problem to find the optimal decision of the manufacturer. Furthermore, some features of the optimal decision are specified and proved and two heuristic algorithms are proposed to solve the problem. Finally, the solutions are evaluated and the effectiveness of the algorithms are demonstrated.

The rest of the paper is organized as follows. In Section 2, the relevant literature is reviewed. In section 3, the problem is defined in detail. The model is formulated in Section 4. In Section 5, numerical examples are described and analyzed. Finally, the conclusion is presented in Section 6.

## 2. Literature review

For the problem of single machine scheduling with the objective of minimizing total number of tardy jobs, which is denoted by  $1 || \sum U_j$  according to three-field notation (Graham (1979)), an algorithm is developed by Moore (1968) that solves the problem in  $O(n \log n)$ . However, when release dates are introduced ( $1 | r_j | \sum U_j$ ), the problem is proved to be strongly NP-hard (Lenstra et al., 1977). Various

models and algorithms have been developed to minimize total number of tardy jobs (Adamu and Adewumi (2014)). Since the subject of this study is more general, in this section, we only review the main studies that consider both machine scheduling and batch delivery costs.

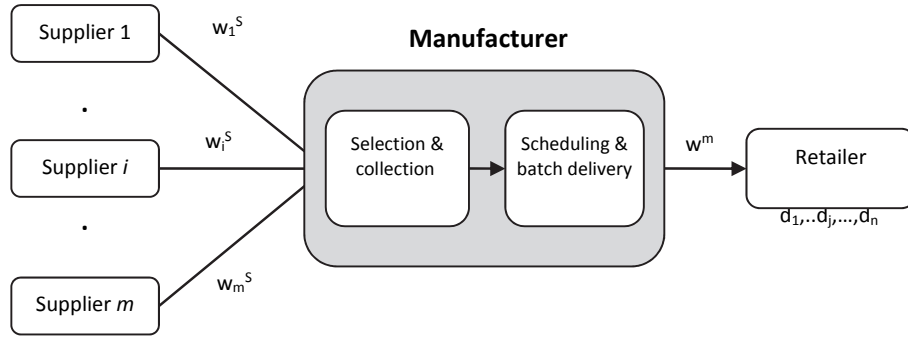
Cheng and Kahlbacher (1993) are believed to be the first who consider machine scheduling with batch delivery costs. Cheng et al. (1996) study single machine scheduling with the objective of minimizing the sum of earliness penalties and batch delivery costs. Lee and Chen (2001) incorporate transportation capacity and transportation times into the machine scheduling problem. Hall and Potts (2003) analyze scheduling and batching problems with three different objectives and present several dynamic programming algorithms to solve them. Pundoor and Chen (2005) study scheduling of a production-distribution system to optimize maximum delivery tardiness and total distribution cost. Chen and Vairaktarakis (2005) present an integrated scheduling model of production and distribution concerning customer service level and total distribution cost. Ji et al. (2007) consider a single machine scheduling and batch delivery system with the objective of minimizing the sum of the total weighted flow time and delivery costs. Mazdeh et al. (2007) introduce a branch-and-bound algorithm for a single-machine scheduling problem with batch delivery to minimize the sum of flow times and delivery costs. Mazdeh et al. (2008) extend this problem by assuming release times for jobs. The objective function of sum of weighted flow times and delivery costs with and without release times are also addressed respectively by Mazdeh et al. (2012). Herrmann and Ullrich (2013) consider machine scheduling and batch delivery with the objective function of sum of tardiness, inventory holding and transportation costs. Rasti-Barzoki et al. (2013) apply a branch-and-bound algorithm to minimize the sum of total weighted number of tardy jobs and delivery costs for a single machine and two-machine flow shop scheduling problem. Selvarajah and Zhang (2014) address the problem for a manufacturer and several customers to minimize the sum of weighted flow time and batch delivery costs. Yin et al. (2015) consider integrated production and batch delivery scheduling for two competing agents seeking to minimize an objective function depending on the completion times of their jobs on a shared single machine. Ahmadizar and Farhadi (2015) present a mathematical model for the problem of single-machine scheduling and batch delivery with due windows in order to minimize the sum of earliness, tardiness, holding, and delivery costs. Finally, other relevant recent works can also be found in Li (2015), Mazdeh and Karamouzian (2014), Tyagi et al. (2013), Yin et al. (2013), Mazdeh et al. (2011), Mazdeh et al. (2012) and Haddad et al. (2012).

All above mentioned works do not consider the issue of sourcing and supplier selection described in the previous section. Furthermore, other research that study supplier selection such as Tadeusz (2011), Razi (2014) and Parhizkari et al. (2013) do not include batch delivery of jobs to customers or machine scheduling issues. This paper tries to fill this gap by incorporation of supplier selection into the integrated machine scheduling and batch delivery problem. Following three-field notation, this problem can be shown as  $1|r_j|(\sum U_j + \sum D_k + \sum \sigma_i)$  where  $D$  and  $\sigma$  refer to delivery costs and supply costs respectively. It should be noted that, the ready times of jobs for processing are also decision variables depending on which suppliers are selected. Since problem  $1|r_j|(\sum U_j)$  is proved to be strongly NP-hard (Lenstra et al. (1977)), the proposed problem which is a generalization of it is also strongly NP-hard. We present a heuristic algorithm (greedy algorithm) and also a local search to solve the problem in addition to mathematically modelling the problem.

### 3. Problem definition

Consider a manufacturer that receives orders from a retailer. Let  $I_j = \{1, 2, \dots, j, \dots, n\}$  denotes the set of retailer orders (jobs) and  $\{d_1, \dots, d_j, \dots, d_n\}$  their corresponding due dates. Each job has a processing time ( $t_j$ ) on the manufacturer machine (facility) and only one job can be processed at a time. The manufacturer should schedule the jobs (orders) so that all jobs meet their due dates. If a job is delivered to the retailer after its due date, it is regarded as a tardy job and the manufacturer incurs a penalty cost

( $\alpha$ ). It is possible to either dispatch a job to the retailer as soon as its processing is completed or wait until some other jobs being ready and send them together as a batch. Each batch delivery costs the manufacturer  $w^m$  and is independent of the number of jobs in the batch. Although, batching decreases the delivery costs, it might lead to tardiness of some jobs (the batches are denoted by  $\{1, 2, \dots, k, \dots, K\}$ ). The manufacturer needs some parts or specific materials to complete each retailer order. These parts can be purchased from some suppliers, denoted by  $I_s = \{1, 2, \dots, i, \dots, m\}$ . Based on the production capacity, the location and the size of jobs being undertaken, every supplier offers a release date for any job (manufacturing parts). Let  $r_{ij}$  denote the release date of manufacturing parts for job  $j$  offered by supplier  $i$  and  $w_i^s$  corresponds to the transportation cost of a job from supplier  $i$  to the manufacturer. Here, the manufacturer needs to first decide from which supplier to purchase parts and second determine the scheduling and sequencing of jobs in its facility and third specify the placing of orders into batches. The decision environment is illustrated in Fig. 1.



**Fig. 1.** The decision context of the manufacturer

To mathematically formulate the decision problem of the manufacturer, the following variables in addition to aforementioned ones are defined.

- $S_j$  Start time of processing job  $j$
- $D_k$  Delivery time of  $k^{th}$  batch to the retailer
- $X_{ij}$  Binary variable takes 1 if job  $i$  is to be processed just before job  $j$  (no other job is processed between them) and 0 otherwise.
- $A_{jk}$  Binary variable takes 0 if job  $j$  is assigned to the  $k^{th}$  batch of the manufacturer delivering to the retailer, 1 otherwise.
- $B_k$  Binary variable takes the value 1 if at least one job is assigned to the  $k^{th}$  batch, 0 otherwise.
- $N_{ij}$  Equals 1 if job  $j$  is assigned to supplier  $i$ , 0 otherwise.
- $U_j$  Equals 1 if job  $j$  is tardy and is equal to 0 otherwise

The manufacturer needs an optimal integrated decision to minimize its costs incurred by the retailer and selected suppliers. The optimal decision can be found by the following formulations:

$$\min TC = \alpha \sum_{j=1}^n U_j + \sum_{i=1}^m \sum_{j=1}^n (w_i^s N_{ij}) + \sum_{k=1}^K w^m B_k \quad (1)$$

subject to

$$\sum_{j=1, j \neq h}^n X_{jh} \leq 1 \quad h = 1, 2, \dots, n \quad (2)$$

$$\sum_{h=1, h \neq j}^n X_{jh} \leq 1 \quad j = 1, 2, \dots, n \quad (3)$$

$$\sum_{j=1}^n \sum_{h=1, h \neq j}^n X_{jh} = n - 1 \quad (4)$$

$$S_j \geq \sum_{i=1}^m r_{ij} N_{ij} \quad j = 1, 2, \dots, n \quad (5)$$

$$S_j \geq S_l + t_l - L(1 - X_{lj}) \quad j, l = 1, \dots, n; l \neq j \quad (6)$$

$$D_k \geq S_j + t_j - L(1 - A_{jk}) \quad j = 1, 2, \dots, n; k = 1, \dots, K \quad (7)$$

$$(L)U_j \geq D_k - d_j - L(1 - A_{jk}) \quad j = 1, 2, \dots, n; k = 1, \dots, K \quad (8)$$

$$\sum_{i=1}^m N_{ij} = 1 \quad j = 1, 2, \dots, n \quad (9)$$

$$\sum_{k=1}^K A_{jk} = 1 \quad j = 1, 2, \dots, n \quad (10)$$

$$\sum_{j=1}^n A_{jk} \leq L(B_k) \quad k = 1, \dots, K \quad (11)$$

$$A_{jk}, B_k, U_j, X_{lj}, N_{ij} \in \{0, 1\} \quad (12)$$

$$D_k, S_j \geq 0 \quad l, j = 1, 2, \dots, n; k = 1, \dots, K$$

The objective function (1) minimizes total cost, which consists of total tardiness costs, total collection costs from suppliers and total dispatching costs to the retailer, respectively. Constraint sets (2) to (4) control the sequence of jobs for processing. Each job is either the first one processed or succeeds another one (constraint (2)) and similarly each job either precedes another job or is the last one processed on a machine (constraint (3)). In a sequence with  $n$  jobs, there are exactly  $(n-1)$  jobs that are succeeded (or proceeded) other jobs which is established by constraint (4). The manufacturer cannot process a job until its material is released by the selected supplier (constraint (5)). The start time of processing a job also needs to be greater than the completion time of its preceding job which is reflected in constraint (6). Note that “ $L$ ” is a sufficiently large number. Constraint (7) ensures that the delivery time of a batch to the retailer is after the completion time of all jobs belonging to that batch. Constraint (8) specifies whether a job is tardy or not. Constraint (9) states that a supplier has to be assigned to a job for collection and similarly constraint (10) assures that a job is assigned to a batch for delivering. Constraint set (11) determines whether a batch is real or not. If at least one job is assigned to a batch then the batch is real and a transportation cost has to be considered in the objective function, while an imaginary (not real) does not influence the objective function and it is mentioned just to model the problem. Finally, constraint sets (12) define the type and boundary of variables. The resulting model is a Mixed Integer Linear Programming (MILP) with  $2n$  continuous variables and  $5(m)(n)$  binary variables. This type of model can be solved with exact algorithms for small size problems. However, for medium and large problems, they are either intractable when solved with full-space approaches or poor solution results. In the following, we propose a greedy algorithm and a local search procedure to cope with the complexity of the problem.

## 4. Heuristics

### 4.1 Structural features

In this part, some properties of the optimal solution of the problem are demonstrated. The results are then applied to the construction and justification of the proposed heuristics. Before presenting the properties, first the following relations are defined for each job ( $j$ ) in a predefined sequence of jobs ( $\delta$ ).

$$LR_j = \min_{i=1, \dots, m} \{r_{ij}\} \quad (13)$$

$$g_j(e) = \begin{cases} \min_{i=1,\dots,m} \{w_i^s | r_{ij} = LR_j\} & LR_j > e \\ \min_{i=1,\dots,m} \{w_i^s | r_{ij} \leq e\} & LR_j \leq e \end{cases} \quad (14)$$

$$h_j(e) = \min_{i=1,\dots,m} \{r_{ij} | w_i^s = g_j(e), i = 1, 2, \dots, n_s\} \quad (15)$$

$$f_j^1 = \frac{d_j}{\sum_{l \in I_j} d_l} \quad (16)$$

$$f_j^2(e) = \frac{t_j + \max(e, h_j(e))}{\sum_{l \in I_j} [t_l + \max(e, h_l(e))]} \quad (17)$$

$$f_j^3(e) = \frac{g_j(e)}{\sum_{l \in I_j} g_l(e)} \quad (18)$$

In the abovementioned functions, the input  $e$  is the earliest time that job  $j$  in sequence  $\delta$  can be processed. It takes the value of zero for the first job and for the rest equals to the completion time of preceded jobs. Eq. (13),  $LR_j$ , determines the minimum release time for a job. The function  $g_j(e)$ , finds the supplier with the least transportation costs for job  $j$  that is planned to start at  $e$  and reports its transportation cost. If all suppliers release job  $j$  after time  $e$ , then this variable finds the supplier with the minimum release time. The function  $h_j(e)$  (15) acts similarly but instead of the transportation costs, it reports the release time of the selected supplier for job  $j$ . The functions  $f_j^1, f_j^2(e)$  and  $f_j^3(e)$  EQ. (16-18) are defined so as to compare jobs to be scheduled at time  $e$  in terms of the earliest due dates, earliest possible completion times and least transportation costs respectively. Note that  $f_j^1, f_j^2(e)$  and  $f_j^3(e)$  are dimensionless.

**Definition 1.** Selection of a supplier based on function  $h_j(e)$  for job  $j$  in sequence  $\delta$  is a procedure such that first the completion time of the preceded jobs of job  $j$  in the sequence is calculated ( $C'$ ) and then the supplier of job  $j$  is determined by the value of  $h_j(C')$ . (Note that  $h_j(C')$  yields the release time of the job which indirectly determines which supplier is selected )

**Proposition 1.** In the optimal decision of the problem, if job  $l$  is placed before job  $j$  in the sequence then the delivery time of job  $j$  is greater or equal to delivery time of job  $l$ .

**Proof** (by contradiction). Consider that the delivery time of job  $j$  is less than job  $l$  in the optimal decision. Hence, the jobs are assigned to different batches namely  $k_l$  and  $k_j$ . Since job  $j$  is assigned to batch  $k_j$ :

$$\left. \begin{array}{l} D_{k_j} \geq C_j \\ l \ll j \Rightarrow C_l < C_j \end{array} \right\} \Rightarrow C_l < D_{k_j} \quad (19)$$

Also, from the assumption we have  $D_{k_j} < D_{k_l}$ . Thus,  $C_l < D_{k_j} < D_{k_l}$ . It means that job  $j$  and all its preceded jobs in batch  $k_l$  can be assigned to batch  $k_j$  without any increase in the objective function (the objective function might be decreased due to reduction of batches). Consequently, the delivery times of job  $l$  and  $j$  become the same which contradicts the assumption.

**Proposition 2.** In the optimal decision, the delivery time of a batch equals to the longest completion time in that batch ( $D_k = \max\{C_j, j \in k\}$  )

**Proof.** It is clear that if  $D_k > \max\{C_j, j \in k\}$ , then we can reduce  $D_k$  so that it equals to  $\max\{C_j, j \in k\}$  to improve the objective function by reducing the number of tardy jobs. Note that  $D_k < \max\{C_j, j \in k\}$  is not possible since according to the definition of a batch, all jobs belongs to a batch must become completed for delivering.

**Proposition 3.** For any job ( $j$ ) in a predefined sequence, if its supplier is assigned based on  $h_j(e)$ , then the completion time of any jobs and therefore makespan ( $C^*$ ) is minimized.

**Proof** (by contradiction). Consider at least for one job (job  $j$ ),  $r_{ij}$  is not equal to  $h_j(e)$  and the completion time of jobs ( $C$ ) is less than ( $C^*$ ). Since  $C < C^*$  then at least one job ( $l$ ) must start sooner in comparison with previous situation. Here, two cases need to be distinguished:

*Case 1.* Job  $j$  is in the first order of the sequence. In this case, the earliest time for starting job  $j$  is  $\min\{r_{ij}\}$  for  $j = 1, \dots, m$ . This value equals to  $h_j(0)$  and therefore it contradicts (note that  $e$  is 0 for the first job).

*Case 2.* Job  $j$  is in the second or greater order. Based on *case 1*, the first job cannot be started sooner and therefore the completion time of the first job ( $C_1$ ) is not changed. The earliest time that the second job can be processed is:

- a) If at least there is a supplier with  $r_{ij} \leq C_1$  equals  $C_1$
- b) If all  $r_{ij} > C_1$  then it is  $\min_{j=1, \dots, m} \{r_{ij}\}$

In other words, the earliest possible time for processing second job is  $\max(C_1, \min_{j=1, \dots, m} \{r_{ij}\})$  which equals to  $h_j(C_1)$  and therefore it contradicts. Thus, the start time of the second job is not changed. Similarly, for the remaining jobs this justification can be applied and therefore no job can be processed sooner in comparison with previous situation which contradicts the assumption.

**Proposition 4.** For two schedule of jobs namely  $\delta_1, \delta_2$  with the same sequence and different suppliers, if for any job  $C_j(\delta_1) \leq C_j(\delta_2)$  then there is always an optimal delivery decision such that the sum of delivery and tardiness cost of  $\delta_1$  is less than or equal to any delivery decision for  $\delta_2$ .

**Proof.** First consider the optimal delivery decision of  $\delta_2$ . According to Proposition 2, for any batch we have  $D_k(\delta_2) = \max\{C_j(\delta_2), j \in k\}$ . We can adopt a delivery decision for  $\delta_1$  similar to optimal delivery decision of  $\delta_2$ , where batching is the same. Since  $C_j(\delta_1) \leq C_j(\delta_2)$  then we have for this decision  $D_k(\delta_1) \leq D_k(\delta_2)$ . In this situation, the number of batches and also material cost are the same as  $\delta_2$ . Due to fact that a job is a tardy when  $D_k > d_j (j \in k)$ , it is found that tardiness cost in  $\delta_2$  is less than or equals to  $\delta_1$ . Consequently, total cost for optimal decision of  $\delta_2$  is less than or equals to  $\delta_1$ .

**Corollary 1.** For a given sequence, assigning suppliers (selecting release time) according to  $h_j(e)$  procedure can yield optimal delivery decision for that sequence.

**Proof.** Proposition (4) states that any supplier selection that yields shorter completion time for each job results in lower delivery costs. Since  $h_j(e)$  procedure has the minimum completion time for all jobs in the sequence (Proposition (3)), therefore it is the best possible selection strategy for minimizing delivery costs of jobs.

**Proposition 5.** For a given set of suppliers (selected suppliers or  $r_{ij}$ ) the best sequence for minimizing sum of tardy jobs is such that all tardy jobs are at the end of the sequence.

**Proof.** Consider that job  $j$  is placed in position  $k$  and it is a tardy job. When this job is moved to the last of the sequence, all jobs between these positions can start earlier and consequently are finished sooner. Only job  $j$ , starts later in comparison with the first sequence and since this job is already late, the number of tardy jobs is not increased or might be decreased. By repeating this operation for all tardy jobs before early jobs we can reach an optimal sequence where all tardy jobs are at the end.

**Proposition 6.** *If there is a job that has the minimum value of  $f_j^2(e)$  for  $e=0$  among all jobs then in the optimal decision of minimizing the idleness time of scheduling (or minimizing the makespan) this job is in the first of schedule.*

**Proof.** It is clear that if any other job is in the first order we can reduce the idleness time of the first processing job by replacing this job with the job that has the minimum value of  $f_j^2(e)$ . Note that  $f_j^2(e)$  for  $e=0$  equals to the earliest release time for that job.

In the next two parts, the heuristic algorithms are presented. Note that the properties of the optimal solution described and proved in this section, propositions (1-6) and Corollary (1), are applied to construct the subsequent solution algorithms.

#### 4.2 Greedy heuristic

A greedy heuristic which is a constructive algorithm starts from scratch (empty solution) for a given optimization problem and constructs a solution by assigning values to one decision variable at a time, until a complete solution is generated (Talbi, 2009). Here, we have to specify the schedule of jobs, the suppliers and also define batches which are strictly related to each other. In the greedy algorithm, first a job for processing is selected from unscheduled jobs based on the value  $f_j^1, f_j^2(e)$  and  $f_j^3(e)$ . Then a supplier is assigned to that job according to  $h_j(e)$  (see definition 1). This procedure is repeated until all jobs are scheduled. Finally, the batching decision is made and the supplies are revised again based on the delivery times of batches. The proposed greedy heuristic (GRA) is as follows,

**Step 1.** Set  $I = \emptyset$ ,  $I_T = \emptyset$ ,  $\bar{I} = \{1, 2, \dots, j, \dots, m\}$ ,  $e = 0$

**Step 2.** While (  $\bar{I} \neq \emptyset$  ) do

2.1 Calculate  $f_j^1, f_j^2(e), f_j^3(e)$  for all  $j \in \bar{I}$

2.2 Select job  $j$  which has the minimum value of  $(f_j^1 + f_j^2(e) + f_j^3(e))$

2.3 Calculate  $h_j(e)$  and  $C_j$

2.4 If  $C_j \leq d_j$  then  $I = I \cup \{j\}$  and  $\bar{I} = \bar{I} \setminus \{j\}$ . Also, update  $e = C_j$  and assign the supplier based on  $h_j(e)$  and  $g_j(e)$

2.3 If  $C_j > d_j$  then  $I_T = I_T \cup \{j\}$  and  $\bar{I} = \bar{I} \setminus \{j\}$ . Also, assign the supplier based on  $h_j(L)$  and  $g_j(L)$  where  $L$  is a large number.

**Step 3.**  $I = I \cup I_T$

**Step 4.** Start from the first job in  $I$ , and add jobs to the first batch until the tardiness cost is greater than  $w^m$ . Repeat this action to batch all jobs.

**Step 5.** If there is a tardy batch, evaluate the batching of the two last batches. If it is beneficial, merge them.

**Step 6.** For  $k=1$  to Number of batches

6.1 Find the idleness time between batch  $k$  and  $(k+1)$  and set it as  $t$

6.2 calculate  $q = \max\{0, \min\{D_k - C_j\}\}$  for any job  $j$  in batch  $i$

6.3 Delay the first job in batch  $i$  by  $\min\{q, t\}$

6.4 Reassign the suppliers for jobs in batch  $j$  based on new start times

**Step 7.** End



Step 1 and step 2 of GRA, try to schedule jobs such that the jobs with shortest processing time and due dates are placed in the first orders and the tardy ones at the end. This work can yield better solutions in terms of the number of tardy jobs and batch delivery costs (Propositions (5-6) and Corollary (1)). Step 4 and 5 adopt the batching rule from proposition 1 and 2 so that the batching decision starts from the first job in the sequence and is completed when the cost of tardiness of jobs in the corresponding batch is greater than  $w^m$ . In step 6, the float of jobs is identified and the cost of suppliers is reduced by delaying the start time of processing of jobs whatever it is possible.

#### 4.3 Local search

Although the greedy heuristic (GRA) can yield high quality solutions in a very short time (according to the computational results), it does not necessarily produce global solutions for any problem since it does not search the whole feasible space. Hence, we present a local search procedure (GRLS) to relatively overcome this defect. The algorithm starts from the solution of the greedy algorithm as initial solution and follows the below steps.

**Step 1.** Set the solution of the greedy heuristic as initial solution and its objective function as  $z^*$

**Step 2.** For  $I=1$  to  $N_T$  ( $N_T$  is the total number of iterations)

2.1 Select two jobs randomly from the sequence

2.2 Interchange the place of jobs and assign suppliers to the new sequence based on  $h_j(e)$  procedure

2.3 Batch the jobs based on Steps 4,5 and 6 of the greedy heuristic

2.4 Calculate the objective function ( $z$ )

2.5 If  $z < z^*$  then replace the new solution with best found solution and  $z^* = z$  else do nothing

2.6 Select a job randomly (job  $j$ )

2.7 Change the supplier of job  $j$  according to  $h_j(L)$  where  $L$  is a large number

2.8 Batch jobs similar to (2.3)

2.4 Calculate the objective function ( $z$ )

2.5 If  $z < z^*$  then replace the new solution with the best found solution and  $z^* = z$  else do nothing

**Step 3.** Report the best found solution and  $z^*$

As mentioned before, the sequence of jobs is a critical decision that affects the objective function considerably. Thus, in GRLS we examine the changes in the sequence of jobs through step 2.1 to step 2.5. Furthermore, the supplier assignment is disturbed by step 2.6 to step 2.8. This procedure can find the situations where deliberately delay of processing a job is beneficial. In the local search procedure we just accept any changes that improve the objective function.

### 5. Computational results

To validate the proposed heuristics and evaluate their effectiveness, it is necessary to build some random test problems. Various test problems in the field of scheduling are presented in the OR library but they are not useful here since our problem is completely different from them. We generate random problems based on Dauzère-Pérès (1995). The processing time for each job ( $p_i$ ) is generated from the uniform distribution  $[1,100]$ . The release time of each job for a supplier ( $r_{ij}$ ) is an integer generated from the uniform distribution  $[0, K_1 n]$ . Also, for each job a supplier is selected randomly ( $r_{ij}^*$ ) and an integer due date  $d_i$  is generated from the uniform distribution  $[r_{ij}^* + p_i, r_{ij}^* + p_i + K_2 n]$ . Two parameters  $K_1, K_2$  are taken randomly in the set  $\{1,5,10,20\}$ . Finally,  $\alpha, w_j^S$  and  $w^m$  is generated randomly from the uniform distribution  $[5,15]$ ,  $[1, \alpha/2]$  and  $[1, \alpha/2]$  respectively.

In our survey, 14 test problems with different sizes are generated where the first six problems are grouped as small-sized and the rest of them are medium and large size problems (the size of problems is based on the number of jobs and the number of suppliers ( $n \times m$ )). We apply a solver to find the global solutions of the mathematical problem to validate the solutions produced by the heuristics for small size problems and also evaluate the quality of solutions for large size problems. It should be noted that due to NP-hardness of the problem, the solver can solve only small size problems and for medium and large ones it is unable to find the global optimum in a limited amount of time and usually results in

poor solutions. Both heuristics are coded in MATLAB 7.7, and the mathematical models have been solved by LINGO 8.0 on a Laptop computer with 2.27 GHz CPU and 4 GB RAM. In Table 1, the problem instances and their corresponding results are listed. For each problem, the local search procedure is run 10 times and the average and the best objective function, the average computational time, and the standard deviation of results are reported. According to Table 1, for small sized problems (1-6), both heuristics generate the global solutions in less than one second. For problem instances 7-10, GRLS still finds the global solutions while GRA either produces global solutions or very close solutions to the optimal. Finally, for large instances (11-14) both heuristics generate very good solutions compared with best found objective of the solver. For instance, the best solution found for problem 14 by the solver is 808 in 30 hours while GRA reaches to objective function 585 (28% less than the solver) in only 0.26 seconds.

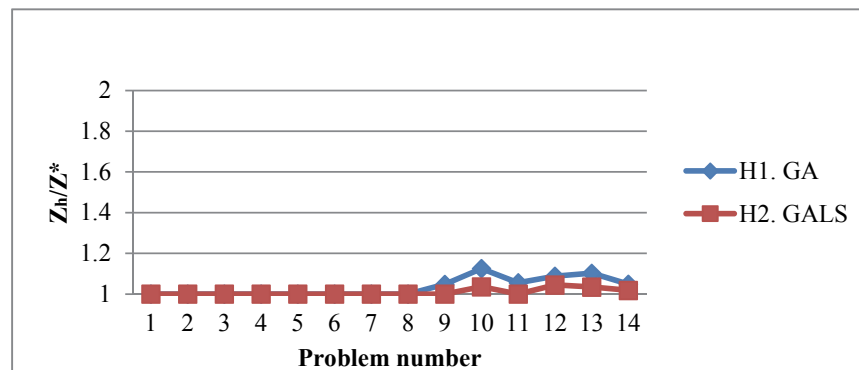
**Table 1**

Detailed computational results comparing the results of the solver and the heuristics

Pro. Num	Data		Solver		GRA		GRLS			
	N	m	Objective	Time(s)	Objective	Time(s)	Avg.	Best	Avg.	Std
1	2	5	32	1	32	0.06	32	32	0.1	0
2	3	4	17	2	17	0.06	17	17	0.15	0
3	3	6	21	1	21	0.06	21	21	0.26	0
4	4	6	56	32	56	0.06	56	56	0.27	0
5	4	10	31	30	31	0.06	31	31	0.4	0
6	5	3	40	300	40	0.06	40	40	0.1	0
7	5	7	34	18000	34	0.07	34	34	0.4	0
8	6	4	42	39600	42	0.06	42	42	0.42	0
9	6	8	21	240	22	0.07	21	21	0.67	0
10	6	10	56	43200	63	0.07	58	58	0.7	0
11	10	10	100*	90000	95	0.08	90	90	2.3	0
12	20	20	215*	90000	161	0.15	154.6	148	15.8	4.03
13	30	30	213*	90000	129	0.12	121	117	52	2.9
14	50	50	808*	90000	585	0.26	568	558	328	4.1

\* Best objective function after 25 hours running the solver. It is not the global solution.

In order to precisely evaluate the performance of the heuristics, measure  $(z_h / z^*)$  is defined in which  $z_h$  refers to the object function of the heuristic and  $z^*$  corresponds to either the optimal solution or the best found solution by the heuristics or the solver (for problems 1-10,  $z^*$  reflects the optimal solution). The minimum value of the measure is 1, which shows that the heuristic has found the optimal or the best solution. The values of the measure greater than 1, which are associated with deviations from the optimal solution.

**Fig. 2.** Values of  $(z_h / z^*)$  for GRA and GRLS

In Fig. 2, the values of the measure for all problems are illustrated. Moreover, in Table 2 the summary of comparing results of the heuristics including the average amounts of the measure and computation times are represented according to the problems categories. The results indicate that both heuristics have no deviations from the global solutions for small size problems. The average deviation for GRA in medium and large size problems are 4% and 7%, respectively while for GRLS, these values

correspond to 0.8% and 2%, respectively. GRLS can reach to better solution compared with GRA but its computation time is greater than GRA. It is concluded that all problem instances can be solved either optimally or very close to optimal by both heuristic algorithms in very short time.

**Table 2**  
Summary of performance evaluation of GRA and GRLS

Size	GRA (Greedy Algorithm)				GRLS			
	$(z_h / z^*)$		CPU time(s)		$(z_h / z^*)$		CPU time(s)	
	Avg	Max	Avg	Max	Avg	Max	Avg	Max
Small (Pr: 1-6)	1	1	0.06	0.06	1	1	0.21	0.4
Medium (Pr: 7-10)	1.04	1.125	0.67	0.07	1.008	1.035	0.54	0.7
Large (Pr: 11-14)	1.07	1.1	0.15	0.26	1.02	1.04	99.5	328

## 6. Conclusion

This paper has incorporated supplier selection issue into the problem of single machine scheduling with batch delivery. The importance of introducing the issue in the original problem is that it has a very significant effect on the sequencing and batching of jobs and the evaluation and selection of suppliers cannot be treated as an independent task. We have presented a mathematical model that formulates the optimal decision of the manufacturer. Since it is a strongly NP-hard problem, we have also proposed a constructive (greedy) heuristic and a local search algorithm. The computational results have shown that the greedy algorithm was very efficient in terms of both quality of solutions and computation time. The obtained results have also indicated that the local search procedure that is combined with the greedy heuristic acts as a complementary tool and helps to find global or near global solutions whenever the greedy algorithm is not able to find that. Our future research will be to extend the suppliers' proposal so that they offer different price for jobs or give a discount when more than one job is purchased. Another possible extension of this paper would be to consider different objective function for scheduling such as total weighted tardiness cost, sum of flow times and sum of completion time.

## Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful and constructive comments that greatly contributed to improving the final version of the paper.

## References

- Adamu, M. O., & Adewumi, A. O. (2014). A survey of single machine scheduling to minimize weighted number of tardy jobs. *Journal of Industrial & Management Optimization*, 10(1), 219-241.
- Ahmadizar, F., & Farhadi, S. (2015). Single-machine batch delivery scheduling with job release dates, due windows and earliness, tardiness, holding and delivery costs. *Computers & Operations Research*, 53, 194-205.
- Chen, Z. L. (2010). Integrated production and outbound distribution scheduling: Review and extensions. *Operations Research*, 58(1), 130-148.
- Chen, Z. L., & Vairaktarakis, G. L. (2005). Integrated scheduling of production and distribution operations. *Management Science*, 51(4), 614-628.
- Cheng, T. C. E., Gordon, V. S., & Kovalyov, M. Y. (1996). Single machine scheduling with batch deliveries. *European Journal of Operational Research*, 94(2), 277-283.
- Cheng, T. C. E., & Kahlbacher, H. G. (1993). Scheduling with delivery and earliness penalties. *Asia-Pacific Journal of Operational Research*, 10, 145-152.
- Dauzère-Pères, S. (1995). Minimizing late jobs in the general one machine scheduling problem. *European Journal of Operational Research*, 81(1), 134-142.
- Detienne, B. (2014). A mixed integer linear programming approach to minimize the number of late jobs with and without machine availability constraints. *European Journal of Operational Research*, 235(3), 540-552.

- Graham, e. a. (1979). Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of Discrete Mathematics*, 5(1), 287-326.
- Haddad, H., Ghanbari, P., & Moghaddam, A. Z. (2012). A new mathematical model for single machine batch scheduling problem for minimizing maximum lateness with deteriorating jobs. *International Journal of Industrial Engineering Computations*, 3(2), 253-264.
- Hall, N. G., & Potts, C. N. (2003). Supply chain scheduling: Batching and delivery. *Operations Research*, 51(4), 566-584+674.
- Herrmann, J., & Ullrich, C. A. (2013). The cost-cutting potential of supply chain scheduling. SSRN: 1829138.
- Ji, M., He, Y., & Cheng, T. C. E. (2007). Batch delivery scheduling with batch delivery cost on a single machine. *European Journal of Operational Research*, 176(2), 745-755.
- Lee, C. Y., & Chen, Z. L. (2001). Machine scheduling with transportation considerations. *Journal of Scheduling*, 4(1), 3-24.
- Lenstra, J. K., Rinnooy Kan, A. H. G., & Brucker, P. (1977). Complexity of machine scheduling problems. *Annals of Discrete Mathematics*, 7, 343-362.
- Li, C.-L. (2015). Improved algorithms for single-machine common due window assignment and scheduling with batch deliveries. *Theoretical Computer Science*, 570, 30-39.
- Mazdeh, M. M., Esfahani, A. N., Sakkaki, S. E., & Pilerood, A. E. (2012). Single-machine batch scheduling minimizing weighted flow times and delivery costs with job release times. *International Journal of Industrial Engineering Computations*, 3(3), 347-364.
- Mazdeh, M. M., Hamidinia, A., & Karamouzian, A. (2011). A mathematical model for weighted tardy jobs scheduling problem with a batched delivery system. *International Journal of Industrial Engineering Computations*, 2(3), 491-498.
- Mazdeh, M. M., & Karamouzian, A. (2014). Evaluating strategic issues in supply chain scheduling using game theory. *International Journal of Production Research*, 1-14.
- Mazdeh, M. M., Sarhadi, M., & Hindi, K. S. (2007). A branch-and-bound algorithm for single-machine scheduling with batch delivery minimizing flow times and delivery costs. *European Journal of Operational Research*, 183(1), 74-86.
- Mazdeh, M. M., Sarhadi, M., & Hindi, K. S. (2008). A branch-and-bound algorithm for single-machine scheduling with batch delivery and job release times. *Computers and Operations Research*, 35(4), 1099-1111.
- Moore, J. M. (1968). An n Job, One Machine Sequencing Algorithm for Minimizing the Number of Late Jobs. *Management Science*, 15(1), 102-109.
- Parhizkari, M., Amirib, M., & Mousakhani, M. (2013). A multiple criteria decision making technique for supplier selection and inventory management strategy: A case of multi-product and multi-supplier problem. *Decision Science Letters*, 2(3), 185-190.
- Pundoor, G., & Chen, Z. L. (2005). Scheduling a production-distribution system to optimize the tradeoff between delivery tardiness and distribution cost. *Naval Research Logistics*, 52(6), 571-589.
- Rasti-Barzoki, M., Hejazi, S. R., & Mazdeh, M. M. (2013). A Branch and Bound Algorithm to Minimize the Total Weighed Number of Tardy Jobs and Delivery Costs. *Applied Mathematical Modelling*, 37(7), 4924-4937.
- Razi, F. (2014). A supplier selection using a hybrid grey based hierarchical clustering and artificial bee colony. *Decision Science Letters*, 3(3), 259-268.
- Selvarajah, E., & Zhang, R. (2014). Supply chain scheduling at the manufacturer to minimize inventory holding and delivery costs. *International Journal of Production Economics*, 147, Part A, 117-124.
- Tadeusz, S. (2011). *Scheduling in Supply Chains Using Mixed Integer Programming*. Hoboken, New Jersey: John Wiley & Sons, Inc.
- Talbi, E.-G. (2009). *Metaheuristics: From Design to Implementation*. Hoboken, New Jersey: John Wiley & Sons.
- Tyagi, N., Abedi, M., & Varshney, R. G. (2013). Single machine scheduling in a batch delivery system with fixed delivery dates. *International Journal of Engineering and Technology*, 5(4), 3484-3488.
- Yin, Y., Cheng, T. C. E., Wu, C. C., & Cheng, S. R. (2013). Single-machine common due-date scheduling with batch delivery costs and resource-dependent processing times. *International Journal of Production Research*, 51(17), 5083-5099.
- Yin, Y., Wang, Y., Cheng, T., Wang, D.-J., & Wu, C.-C. (2015). Two-agent single-machine scheduling to minimize the batch delivery cost. *Computers & Industrial Engineering*.