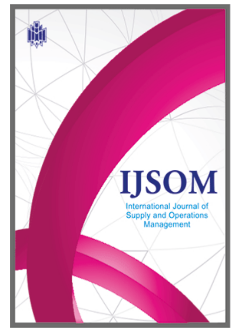
International Journal of Supply and Operations Management

IJSOM

May 2016, Volume 3, Issue 1, pp. 1215-1233
ISSN-Print: 2383-1359
ISSN-Online: 2383-2525
www.ijsom.com



Modeling the hybrid flow shop scheduling problem followed by an assembly stage considering aging effects and preventive maintenance activities

Seyed Mohammad Hassan Hosseini[a]

[a] *Department of industrial engineering and management, Shahrood university of technology, Shahrood, Iran*

Abstract

Scheduling problem for the hybrid flow shop scheduling problem (HFSP) followed by an assembly stage considering aging effects and additional preventive and maintenance activities is studied in this paper. In this production system, a number of products of different kinds are produced. Each product is assembled with a set of several parts. The first stage is a hybrid flow shop to produce parts. All machines can process all kinds of parts in this stage but each machine can process only one part at a time). The second stage is a single assembly machine or a single assembly team of workers. The aim is to schedule the parts on the machines and assembly sequence and also determine when the preventive maintenance activities get done in order to minimize the completion time of all products (makespan). A mathematical modeling is presented and its validation is shown by solving a small scale example. This problem has been proved to be strongly NP-hard, then in order to solve the problem in medium and large scales, four heuristic algorithms is proposed based on the Johnson's algorithm. The numerical experiments are used to run the mathematical model and evaluate the performance of the proposed algorithms.

Keywords: Scheduling; hybrid flow shop; assembly; aging effects; preventive maintenance activities.

Corresponding author email address: sh.hosseini@shahroodut.ac.ir

1. Introduction

One of the most important tasks in manufacturing systems and supply chain management is how to schedule arriving jobs such that some criteria are satisfied (Hentsch et al., 2011), (Mokhtari, Dadgar, 2015), (Elbounjimi, 2015). In particular, manufacturing of almost all items may be modeled as a two-stage assembly scheduling problem (Allahverdi A, Al-Anzi 2009). In these production systems, usually there is a fabrication stage in which the parts or components are processed independently, and an assembly stage in which the components are joined together into the complete products. Fabrication stage can be a single machine, parallel machines, flow shop, hybrid flow shop or job shop. Assembly stage usually is as a single stage or flow shop. The main criterion for this problem is the minimization of the maximum job completion time (Koulamas et al., 2001). This production system has many applications in industry, and hence has received an increasing attention of researchers recently (Lee et al., 1993; Allahverdi et al., 2009; Naderi-Beni et al., 2012). Lee et al. (1993) described an application of this production system in a fire engine assembly plant in 1993. After publishing the first paper about 3-machine assembly-type flow shop scheduling problem by Lee et al. in 1993, the two-stage production system has received considerable attention from researchers during the last two decades. Potts et al. (2009) described another application of this system in personal computer manufacturing. Allahverdi et al. (2009) say that in particular, manufacturing of almost all items may be modeled as a two-stage assembly scheduling problem including machining operations and assembly operations.

According to the above mentioned explanations, despite the importance of two stage production system, generally scheduling for parts machining and planning for assembly operations have been studied independently (Yokoyama et al., 2005). On the other hand, these two stages have impact on influence each other and so, studying these two stages separately may distract the production planning from the ideal goals.

These two stages of production systems are studied in this paper integrally. The first stage is considered as a two stage hybrid flow shop, and the second stage assumed as a single stage or machine. Also the aging effects and additional preventive and maintenance activities for the hybrid flow shop is considered. The hybrid flow shop is a generalization of the classical flow shop in which there are parallel machines for some operations (Blazewicz et al., 2007; Pinedo 2008; Quadt et al., 2007; Ying et al., 2006). In this system, usually there are a set of *n* jobs that must be processed in a series of *m* stages in which there is at least one stage with more than one machine.

The hybrid flow shop that also called flexible flow shop, compound flow shop, and multi-processor flow shop is a generalization of the flow shop in such a way that every job can be processed by one among several parallel machines at each stage (Blazewicz, Ecker, Pesch, Schmidt, Weglarz, (2007), (Quadt, Kuhn, 2007), (Pinedo, 2008). According to literature review there are a number of variants of hybrid flow shop but all of them have several characteristics in common (Ruiz, Vazquez-Rodriguez, 2010) as bellow:

1. The number of processing stages (k) is at least 2.

2. Each stage k has $M^{(K)}$ machines in parallel format and $M^{(K)} \geq 1$ for all stages and $M^{(K)} > 1$ for at least on stage.

3. All jobs are processed following the same production flow: stage 1, stage 2. . . Stage *K*. A job might skip any number of provided stages, but it is processed in at least one of them.

4. Each job j requires a processing time P_{jk} at stage *k*.

In the standard form of the HFS problem all jobs and machines are available at time zero, machines at a given stage are identical, any machine can process only one operation and any job can be processed by only one machine at a time. Besides that, in this paper setup times are (supposed to be) negligible, preemption is not allowed, and the capacity of buffers between stages is unlimited.

Lee et.al did the first study in assembly-type scheduling problem in 1993 (Lee et. Al, 1993). Their considered problem was a two stage assembly flow-shop scheduling with makespan objective function. They considered that each product is assembled from two types of parts. The first component of each product must be processed on machine A and the second component is processed on machine B. Finally, the third machine assembles the two parts into a product. They proved that the problem is strongly NP-complete and identified several special cases of the problem that can be solved in polynomial time and suggested a branch and bound solution and also three heuristics. After that, Potts et al. (1995) extended the problem to the case of multiple fabrication machines in which there are m machines and one machine at the first and second stages, respectively. They developed a heuristic algorithm with a worst-case ratio bound to minimize the makespan. Hariri and Potts (1997) also studied the same problem as Potts et al. (1995) and proposed a branch and bond algorithm to solve it. Cheng and Wang (1999) studied the problem of scheduling the fabrication and assembly of components in a two-machine flow shop so as to minimize the makespan. In their considered problem, each job consists of two components. A component unique to each product and a component common to all products. Both the unique and the common components are fabricated on the first machine. The assembly operations are performed on the second machine, and can only begin when both components for the job are available. They identified several properties of an optimal solution to the problem, and some polynomial solvable special cases were introduced.

Yokoyama (2001) studied a hybrid scheduling for the production system including parts machining and assembly operations. In his study several products of different kinds are considered and their parts are manufactured in a flow shop. Each product is produced by hierarchical assembly operations from the parts. The parts are assembled into the first sub-assembly, and several other parts and the first sub-assembly are assembled into the second sub-assembly, and this hierarchical assembly structures continue until the final product is obtained in the last stage. The aim is to minimize weighted sum of completion time of each product. Decision variables are the sequence of products to be assembled and the sequence of parts to be processed. He introduced a branch and bound with two lower bounds that can solve problems to the extent of 10 products and 15 parts.

Sun and et al (2003) studied 3-machine, assembly-type flow shop scheduling. They tell that this problem has been proved strongly NP-complete and so proposed a series of heuristic algorithms based on the basic idea of Johnson’s algorithm and Gupta’s idea. The heuristic algorithms can solve all of the worst cases which cannot be solved by the existing heuristic.

Yokoyama and Santos (2005) considered flow-shop scheduling with assembly operations. In their models, several products of different kinds are ordered and parts are manufactured. Each part for the products are processed on machine M_1 (the first stage) and then processed on machine M_2 (the second stage). Then each product is processed e.g., joining parts into the product by one assembly operation at assembly stage M_A (the third stage). They developed a solution procedure to obtain an ε_optimal solution based on a branch-and-bound method.

Allahverdi and Al-Anzi (2009) studied a two-stage assembly scheduling problem where there are m machines at the first stage and an assembly machine at the second stage. In their model the setup times are treated as separate from processing times. They prove that this problem is NP-hard, and therefore present a dominance relation and propose three heuristics that are a hybrid tabu search, a self-adaptive differential evolution (SDE), and a new self-adaptive differential evolution (NSDE). They show that the NSDE is the best heuristic for the problem even if setup times are ignored.

Al-Anzi and Allahverdi (2009) also considered the same problem where setup times are ignored. They prove that this problem is NP-hard, and therefore propose heuristics based on tabu search (Tabu), particle swarm optimization (PSO), and self-adaptive differential evolution (SDE) along with the earliest due date (EDD) and Johnson (JNS) heuristics to solve the problem. Computational experiment reveals that both PSO and SDE are superior to tabu. Moreover, it is statistically shown that PSO performs better than SDE.

Most production scheduling systems assumed that a machine is continuously available during the planning horizon. However, in a real production system, the machine may not be available because of preventive maintenance, tool changes, or breakdowns. Therefore, one of the most important factors in production systems is the depreciation effects and preventive maintenance activities that contain visit and inspection, tuning, replacement, and repair that are done periodically according to the plan (Berrichi, Amodeo, Yalaoui, Chatelet, Mezghiche, 2008). Another important parameter is time processing that usually assumed to be a fixed value. While the time process of a job depends on its sequence position that is called learning effect or aging effect (Bachman, Janiak, 2004). So, scheduling problems with aging (deterioration or fatigue) effects have been extensively studied over two decades in various machine environments and performance measures (Chou-Jun, 2013). The first study on aging effect was done by Moshio (2001). Yang et al. (2012) showed the aging effect in scheduling as below:

$$p_{ir} = p_i \times r^{\propto}$$

While p_i is the process time of the *i*th job in normal condition and p_{ir} is process time of *i*th job in rth position. According to the above formula, by the increment of the process position, the process time will increase too.

Another approach for considering the aging effect on time processing is linier as below:

$$p_{ir} = p_i + b \times r$$

For a complete list of studies, the reader may refer to the comprehensive surveys by Schmidt (2000) and Ma et al. (2010). Also Due to the practical experience in production systems, scheduling with considerations of the learning, aging, or deteriorating effects has been one of the most popular topics among researchers in recent years. The actual processing times of jobs may vary due to the learning, aging, or deteriorating effects. For details on this stream of research, the reader may refer to the comprehensive surveys by Alidaee and Womer (1999), and Cheng et al. (2004).

According to literature review, no study has been done on HFS problem followed by an assembly stage considering the aging effects on the process time and also preventive maintenance activities. So the contribution of this study is considering the learning effect and aging effect on process time in scheduling for the HFS problem followed by an assembly stage in order to near to the practical condition.`

The introduced problem in this study is strongly NP-Hard because the hybrid flow shop scheduling problem restricted to two processing stages, even in the case when one stage contains two machines and the other one a single machine, is NP-hard (Ruiz, Vazquez-Rodriguez 2010). Then it is obvious that this problem with a more complex structure such as considering aging effects and preventive maintenance activities is NP-hard too.

The rest of this paper is organized as follows: In section 2, the problem with an application are described completely and the Mathematical model is presented. The heuristic solving approach is presented in section 3. Design of test problems is described in section 4. After that, comparisons and analysis of the result is done in section 5. Also in section 6, concluding remarks and summary of the work are given and directions for the future research are offered.

2. Problem description

The problem considered in this study can be formally described as follows: several products of different kinds are ordered to be produced. Each product needs a set of components (parts) to complete and the parts are processed on a two stage hybrid flow shop. In each stage of hybrid flow shop there are different number of identical machines. After hybrid flow shop, there is an assembly stage that joins the set parts of a product into the product. Aging effects of the hybrid flow shop machines and preventive maintenance activities is considered for this problem. The aim is to determine sequence of products to be assembled and also scheduling the parts and assigning them to machines in each stage of hybrid flow shop to be fabricated in order to minimize the completion time of all products.
This production system has many applications in industries. Figure 1 shows an application case of this problem in body making of car manufacturing industry. As it is shown, a car making manufactory generally contains the units of production engine, chassis and body. The unit of body making includes press shop, assembly and painting. The press shop that produce some parts such as doors and roofs has usually a flow shop or hybrid flow shop format.

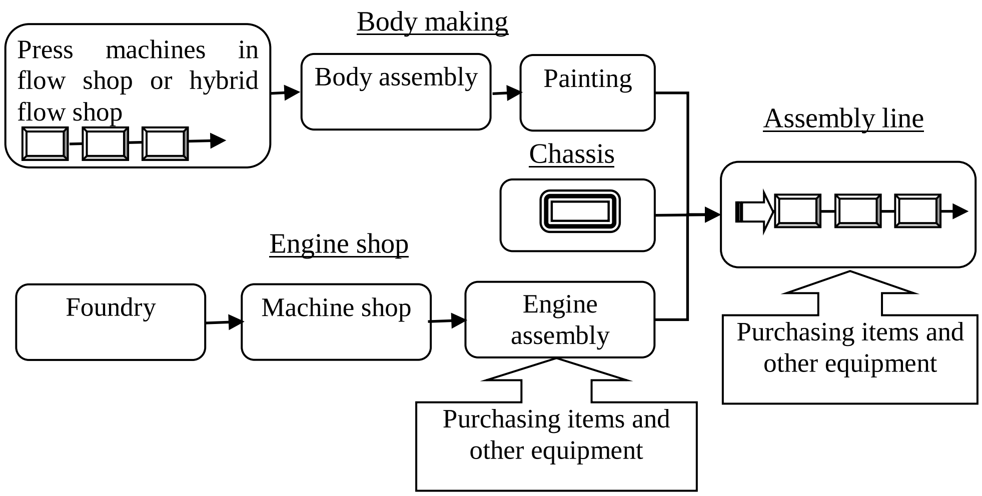


Figure 1. A car making manufactory in general

Figure 2 shows a schematic view of a hybrid production system that contains a flexible flow shop followed by an assembly section. The inputs contain raw material; parts or unfinished products are processed on hybrid flow shop stage. When the set part of a product are complete, they are put together at assembly stage. Typically, buffers are located between stages to store intermediate products and it is supposed that there is no limitation in buffer storages (Quadt, Kuhn, 2007), (Fattahi, Hosseini, Jolai, Tavakoli-Moghadam, 2014). The number of machines at hybrid flow shop stages is free and it can be unequal at two stages.

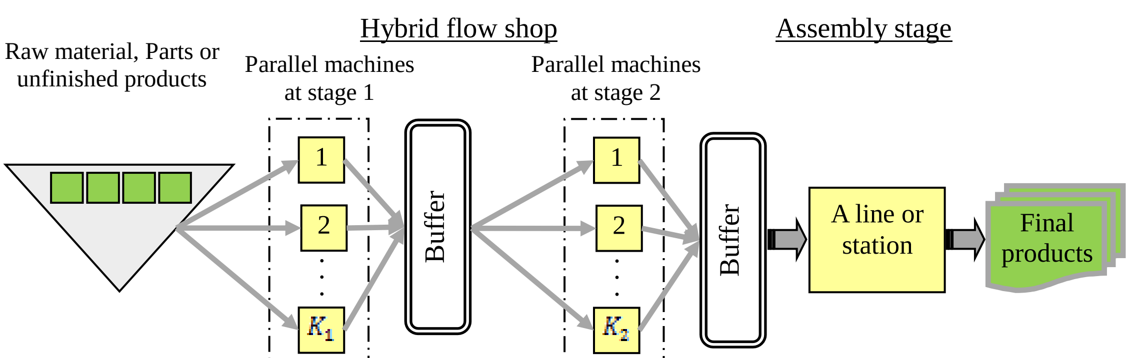


Figure 2. A schematic view of the considered problem

2.1. Notations

We introduce the following indexes for this problem:

h Product index (h = 1, 2, . . ., H)

i, *j* Part index (*i*, *j* = 1, 2, . . ., n)

i_h Set of indices of parts for product *h* ($i_h = 1,2,\ldots,n_h$)

r Position index (*r* = 1, 2, . . ., n)

g Preventive and maintenance activities index (g = 0, 1, . . ., n-1), the zero is a hypothetical value for start of each machine

l Stage index (l=1,2)
K_l Number of parallel machines in stage l
k Machine index,

Parameters of the considered problem are as follows:
P_{li} Processing time of machining operation for part i in stage l (l=1,2)
A_h Assembly time of product h
t_{kl} Preventive and maintenance time for the machine k in stage l
a_{kr} Depreciation rate of machine k for processing parts in position r
M A very big and positive amount

Also the variables of the proposed model are as follows:

x_{ijkl} 1, if job j is processed directly after job i on machine k at stage l, 0 otherwise,
x_{0ikl} 1, if job i is the first job on machine k in stage l, 0 otherwise,
x_{i0kl} 1, if job i is the last job on machine k in stage l, 0 otherwise,
y_{ijrgkl} 1, if gth preventive and maintenance activity is done on machine k at stage l after job i and before job j in position r , 0 otherwise,
z_{gkl} 1, if gth preventive and maintenance activity is done on machine k in stage l, 0 otherwise,
$C_j^{(l)}$ Completion time of job j in stage l,
F_h Completion time of the component parts of the hth product when they are ready to be assembled
$S_{h'h}$ 0, if all parts of product h' is ready to assemble before the parts of product h, a positive amount otherwise,
C_h Total assembly operation time of the product h

The problem is to decide the sequence of products and their parts, and the objective function of problem is expressed as:
$Z = max(C_h)$

2.2. Assumptions

(1) All parts are available at time zero.
(2) The parallel machines at a stage are uniform. each product has a specific machining operation time and it is the same for all parallel machines at the stage
(3) If product h is going to be assembled before product h', then, on each stage, processing of any parts of product h' doesn't start before starting the processing of all parts for product h.
(4) Assembly operation for a product will not start until all parts of it is completed.
(5) When assembly operation of a product is started, it doesn't stop until completed (no preemption is allowed)
(6) There is no limitation in buffer storages
(7) The transportation times are ignored
(8) Depreciation rate of all machines is known and fixed
(9) The aging effect depends on the number of parts that are processed by the machine

2.3. Mathematical modeling

Based on the present problem and notations, a mathematical formulation for the problem is presented as follows:

$$\text{Min } Z = (C_{max}) \quad (1)$$

Subject to:

$$\sum_{i=0}^{n} x_{ijkl} = 1 \qquad j=1,2,3,\dots,n \quad l=1,2 \quad (2)$$

$$\sum_{j=0}^{n} x_{ijkl} = 1 \qquad i=1,2,3,\dots,n \quad l=1,2 \quad (3)$$

$$\sum_{j=0}^{n} x_{ijkl} \le 1 \qquad i=0,1,2,3,n \quad l=1,2 \quad k = 1,2,\dots,K_l \quad (4)$$

$$\sum\nolimits_{i=0,i\ne h}^{n} x_{ihkl} - \sum_{j=0,j\ne h}^{n} x_{hjkl} = 0 \qquad h=1,2,3,\dots,n \quad l=1,2 \quad k = 1,2,\dots,K_l \quad (5)$$

$$\sum_{j=1}^{n}\sum_{j=1}^{n} y_{ijrgkl} \le 1 \qquad g=0,1,2,\dots,n\text{-}1 \quad l=1,2 \quad k = 1,2,\dots,K_l \quad i \ne j \quad (6)$$

$$\sum_{j=1}^{n}\sum_{j=1}^{n} y_{ijrgkl} \le z_{gkl} \times M \qquad g=0,1,2,\dots,n\text{-}1 \quad l=1,2 \quad k = 1,2,\dots,K_l \quad i \ne j \quad (7)$$

$$z_{gkl} \le z_{g-1kl} \qquad g=1,2,\dots,n\text{-}1 \quad l=1,2 \quad k = 1,2,\dots,K_l \quad (8)$$

$$\sum_{j=1}^{n}\sum_{j=1}^{n} y_{ij1gkl} = z_{gkl} \qquad g=0,1,2,\dots,n\text{-}1 \quad l=1,2 \quad k = 1,2,\dots,K_l \quad i \ne j \quad (9)$$

$$C_j^{(2)} \ge C_j^{(1)} + P_{2j} \times (1 + r \times a_{kr}) + t_{kl} - (1 - y_{ijrgkl}) \times M \qquad i,j,r=1,2,\dots,n \quad l=2 \quad k = 1,2,\dots,K_2 \quad i \ne j \quad g=0,1,2,\dots,n\text{-}1 \quad (10)$$

$$C_j^{(l)} \geq C_i^{(l)} \times (1 + r \times a_{kr}) + \left(\sum_{k=1}^{K_l} x_{ijkl} - 1\right) \times M + t_{kl} - (1 - y_{ijrgkl}) \times M \quad \begin{array}{l} i, j, r = 1,2, \ldots, n \quad l = 1,2 \\ k = 1, 2, \ldots, K_l \\ i \neq j \\ g = 0, 1, 2, \ldots, n-1 \end{array} \tag{11}$$

$$F_h \geq C_j^{(2)} \quad \begin{array}{l} \forall j \in \{J_h\} \\ h = 2,3,4,\ldots,H \end{array}, \tag{12}$$

$$C_h \geq F_h + A_h \quad h = 1,2,3,\ldots,H \tag{13}$$

$$C_h \geq C_{h'} + A_h - S_{h'h} \times M \quad h, h' = 1,2,3,\ldots,H \tag{14}$$

$$S_{h'h} \geq F_{h'} - F_h \quad h, h' = 1,2,3,\ldots,H \;,\; h' \neq h \tag{15}$$

$$C_h \leq C_{max} \quad h = 1,2,3,\ldots,H \tag{16}$$

$$x_{ijkl}, y_{ijrgkl}, z_{gkl} \in \{0, 1\} \quad \begin{array}{l} i, j, r = 1,2, \ldots, n \quad l = 2 \\ k = 1, 2, \ldots, K_2 \\ i \neq j \\ g = 0, 1, 2, \ldots, n-1 \end{array} \tag{17}$$

$$C_j^{(l)}, F_h, C_h, S_{h'h}, \geq 0 \tag{18}$$

In the mathematical model, the makespan minimization aspect of the problem is expressed by equation (1). Constraints (2), (3), (4), and (5) ensure that each part is processed precisely once at each stage. In particular, constraint (2) guarantees that at each stage *l* for each part *j* there is a unique machine such that either *j* is processed first or after another job on that machine. The inequalities (3) imply that at each stage *l* for each part *i* there is a unique machine such that either *i* is processed at the end or before another job on that machine. The inequalities (4) imply that at each stage there is a machine on which a part has a successor or is the last processed one. Finally, at each stage for each part there is one and only one machine satisfying both of the two previous conditions by (5).

Constraints (6) show that in each position just one part can be placed on a machine at stage *l*. constraint (7) guarantees that no process will be done until the necessary preventive and maintenance activities are carried out. Constraint (8) shows that the preventive and maintenance activities are done respectively. Constraint (9) implies that after a preventive and maintenance activity, a process is done.

Constraints (10) and (11) take care of the completion times of the parts at stages 1, 2. Inequalities (10) ensure that the completion times C_i^l and C_j^l of parts *i* and *j* which are scheduled consecutively on the same machine respect the predetermined order. Inequality (11) implies that parts go through the stages in the right order, i.e. from stage 1 to stage 2.

Inequalities (12) take care of the start times of the products at assembly stage. The inequalities (13, 14, 15, and 16) express the completion time of products. Inequalities (14, 15 and 16) ensure that the completion time of product h and h' (which are) scheduled consecutively at the assembly stage respect the predetermined order. The constraint that the make span is not smaller than the completion time of each product is expressed by constraints (16). Finally, the last two constraints specify the domains of the decision variables.

3. The proposed solving approach

The decision variables in HFS problem with an assembly stage are the sequence of products to be assembled and assigning the parts to machines in each stage, for all products to be processed. Thus this problem consists of two sub-problems. The first sub-problem is determining the sequence of products to assemble and the second is assigning the parts of product h to a machine in each stage (for h=1, 2, …, H).

According to assumption 3, if product h is going to be assembled before product h', then, at each stage, processing of any part of product h' doesn't start unless all parts of product h are started to be processed. This assumption is considered for many reasons such as reducing the inventory costs, risk of transportations damage and also to facilitate the material flows.

According to these sub-problems, we introduce two steps of solving:

step 1: Find a sequence for product by extension of Johnson algorithm using algorithms *A, B, C, D*.

step 2: Determine the sequence and schedule for the parts of any product using algorithm NEH.

3.1. Sequencing the product

In this section, the Johnson algorithm is extended and used to determine the sequence of the products. So we assume the hybrid flow shop as first stage (or first machine) and assembly as second stage (or second machine). Then the Johnson extended algorithm is used to determine the sequence of products.

The process time of the second stage is assembly time of products and it is shown by A_h (that means assembly time of product h).

In the hybrid flow shop stage, the process time of a product will be computed by four equations presented below. That means the process time of the first stage consists of process operation of components in hybrid flow shop and it is calculated for any product h. This time is shown by PT_h (that is the process time of the set parts for product h) and it will be computed by four algorithms, named *A*, *B*, *C* and *D* as (19), (20), (21), and (22).

Algorithm *JI*: $$PT_h = \sum_{l=1}^{2} \sum_{j \in J_h} P_{lj} \qquad \text{For } h=1,2,\dots,H \quad (19)$$

Algorithm *JII*: $$PT_h = \frac{\sum_{l=1}^{2} \sum_{j \in J_h} P_{lj}}{n_h} \qquad \text{For } h=1,2,\dots,H \quad (20)$$

Algorithm *JIII* $$PT_h = \sum_{l=1}^{2} \frac{\sum_{j \in J_h} P_{lj}}{min(n_h\,,\ K_l)} \qquad \text{For } h=1,2,\dots,H \quad (21)$$

Algorithm *JIV*: $$PT_h = \max_{j \in J_h} \left\{ \max\left(\sum (P_{1j}+P_{2j})\right), \left(\frac{\sum P_{1j}}{K_1} + \min(P_{2j})\right), \left(\frac{\sum P_{2j}}{K_2} + \min(P_{1j})\right) \right\} \quad \text{For } h=1,2,\ldots,H \quad (22)$$

After computing the PT_h based on four presented algorithms, the problem will be solved using Johnson algorithm to determine the sequence of products. This procedure is as below:

1. Determine the PT_h according to equations (15), (16), (17) and (18) and A_h as shown in problem for all products.
2. Suppose $U = \{h \in H | PT_h < A_h\}$ and $V = \{h \in H | PT_h \geq A_h\}$
3. Sort the set of U in PT_h and the set of V in A_h into non-decreasing and non-increasing orders respectively
4. Determine the sequence of products according to the set of U and V after that

3.2. Scheduling the parts

After sequencing the products, in second step the parts of any product will be scheduled to be processed by the algorithm NEH. According to the algorithm *NEH* (Nawaz et al., 1983), the parts are sorted in non- increasing order of their total processing times. This scheduling is done for the first stage of the hybrid flow shop. Obviously for the second stage of the hybrid flow shop, processing will begin with the part of the considered product that is ready earlier.

4. Experimental design

In order to run the mathematical model and evaluate the proposed approach for solving the large scale problems, test problem introduced by Fattahi et al. (2013). They considered three types of problems as shown in Table 3. The problems type 1 (A) present the situation that hybrid flow shop stage is bottleneck, in problems type 2 (B) the assembly stage is bottleneck and in problems type 3 (C) there is a balance condition between two stages. These three types of problems are generated by setting the number of parts. Also the number of machines at both of two stages and number of jobs (products) are changed at each type of problem to have various problems as shown in Table 3.

Suppose that $[P_L, P_U]$ is the range of the number of parts. That is P_L is the lower limit and P_U is the upper limit of the number of parts. Also suppose that $[T1_L, T1_U]$ is the range of processing time of parts in the first stage of hybrid flow shop, $[T2_L, T2_U]$ is the range of processing time of parts in the second stage of hybrid flow shop, and $[Ah_L, Ah_U]$ is the range of assembly time of products. We define index I as below to identify type of the problems:

$$I = \left[H \times \frac{Ah_L + Ah_U}{2}\right] \Big/ \max\left\{\left[H \times \frac{P_L+P_U}{2} \times \frac{T1_L + T1_U}{2} / K_1\right], \left[H \times \frac{P_L+P_U}{2} \times \frac{T2_L + T2_U}{2} / K_2\right]\right\}$$

Because the range for number of parts and their processing times and also the assembly time of products are uniform it is clear that:

If	$I < 1$	Then	The hybrid flow shop stage will be bottleneck
If	$I > 1$	Then	The assembly stage will be bottleneck
Else($I \approx 1$)		Then	There will be balance condition between two stages

Hence we have generated three types of problems as shown in table 1.

Table 1. The batches of problems

Problems types	A	B	C
Interval of index *I*	$I < 1$	$I > 1$	$I \approx 1$
bottleneck	the hybrid flow shop will be bottleneck	the assembly stage will be bottleneck	there will be balance condition

Also, four categories of problems based on the number of products are considered (note: the number of products doesn't change the bottleneck). We consider the number of product in a range between 10 and 150 (10, 50, 100 and 150), the range of process time at stages 1, 2 of hybrid flow shop between [0,100] for all parts and assembly time between [100,300] for all product. The other data is shown in table 2.

Table 2. parameters of the test problems

Scale of Problem	*H*	*n*	k_1	k_2	p_{i1}	p_{i2}	A_h	problem Type
small	10	[2 , 10]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	10	[2 , 10]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	10	[2 , 10]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	10	[2 , 10]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	A
	10	[2 , 10]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	A
	10	[4 , 12]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	10	[4 , 12]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	10	[4 , 12]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	10	[4 , 12]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	C
	10	[4 , 12]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	C
	10	[6 , 16]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	10	[6 , 16]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	10	[6 , 16]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	10	[6 , 16]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	B
	10	[6 , 16]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	B
medium	50	[2 , 10]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	50	[2 , 10]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	50	[2 , 10]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	50	[2 , 10]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	A
	50	[2 , 10]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	A
	50	[4 , 12]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	50	[4 , 12]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	50	[4 , 12]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	50	[4 , 12]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	C
	50	[4 , 12]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	C
	50	[6 , 16]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	50	[6 , 16]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	50	[6 , 16]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	50	[6 , 16]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	B

Table 2. Continued

Scale of Problem	H	n	k_1	k_2	p_{i1}	p_{i2}	A_h	problem Type
	50	[6 , 16]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	B
large	100	[2 , 10]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	100	[2 , 10]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	100	[2 , 10]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	100	[2 , 10]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	A
	100	[2 , 10]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	A
	100	[4 , 12]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	100	[4 , 12]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	100	[4 , 12]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	100	[4 , 12]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	C
	100	[4 , 12]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	C
	100	[6 , 16]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	100	[6 , 16]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	100	[6 , 16]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	100	[6 , 16]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	B
	100	[6 , 16]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	B
	150	[2 , 10]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	150	[2 , 10]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	150	[2 , 10]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	A
	150	[2 , 10]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	A
	150	[2 , 10]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	A
	150	[4 , 12]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	150	[4 , 12]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	150	[4 , 12]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	C
	150	[4 , 12]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	C
	150	[4 , 12]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	C
	150	[6 , 16]	2	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	150	[6 , 16]	3	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	150	[6 , 16]	4	2	[0 , 100]	[0 , 100]	[100 , 300]	B
	150	[6 , 16]	2	3	[0 , 100]	[0 , 100]	[100 , 300]	B
	150	[6 , 16]	2	4	[0 , 100]	[0 , 100]	[100 , 300]	B

The depreciation rate of machine k for processing part in position r is considered as a Uniform distribution of [10% 15%]. Also the preventive and maintenance time for the machine k in stage l has been considered to follow a Uniform distribution of [200 500].

5. Comparisons of results

The mathematical model was used to solve the small scale problems by GAMS program. The result has been written for the problem that was solved in less than one hour, for the others '*' has been marked. Also the proposed solving approach was coded in MATLAB. The experiments are executed on a Pc with a 2.0GHz Intel Core 2 Duo processor and 1GB of RAM memory. Each problem has been solved by all the algorithms. The results of each problem and algorithm is presented in table 5. These results show that
Performance of algorithm *JIV* is better than the others in terms of objective function. That is because the process time of the parts of each product in hybrid flow shop is more accurate in algorithm *JIV*. Also the run time of algorithm *JI* is the best because its computation is simpler.

Figure 3 shows the result of each algorithm for each kind of problem. According to this result, when the stage 1 (hybrid flow shop) is bottleneck (in problems kind A), the objective function decreases dramatically. The reason is that this situation provides a good flexibility for scheduling the parts. On the other hand, when there is no bottleneck and two stages of production process are balanced in terms of working time, there is the least flexibility for scheduling and therefore the objective function value is maximum.

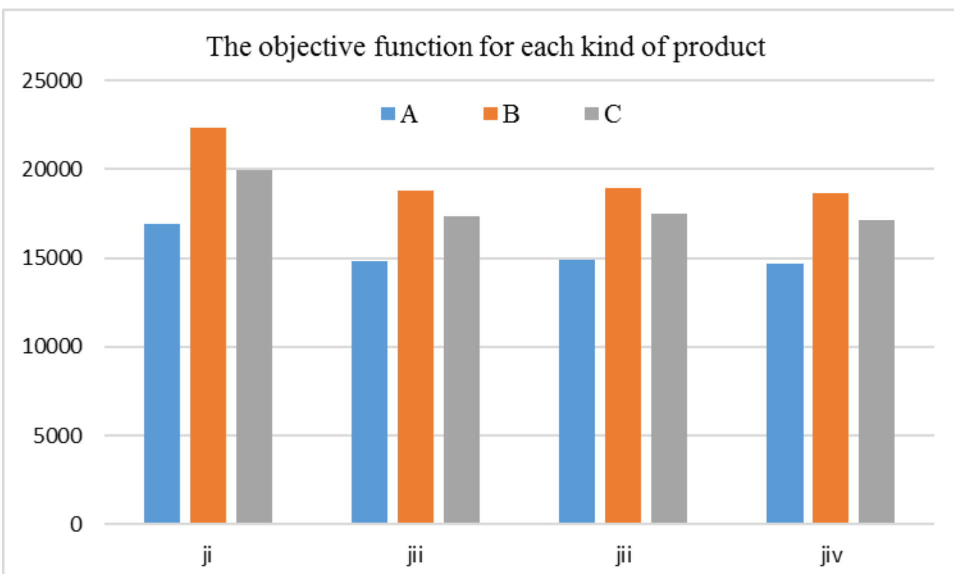


Figure 3. the result of each algorithm for each kind of problem

Table 3. result of the solving of test problems

H	*n*	k_1	k_2	Type	objective function				run time				
					JI	*JII*	*JII*	*JIV*	*JI*	*JII*	*JII*	*JIV*	GAMS
10	[2 , 10]	2	2	A	2908	2698	2775	2664	1.51	2.13	2.12	2.98	1134.2
10	[2 , 10]	3	2	A	2721	2416	2409	2492	1.60	2.22	2.28	3.18	1162.4
10	[2 , 10]	4	2	A	2835	2408	2491	2354	1.62	2.30	2.31	3.20	1168.3
10	[2 , 10]	2	3	A	2824	2688	2794	2565	1.61	2.21	2.27	3.20	1166.5
10	[2 , 10]	2	4	A	2881	2618	2635	2587	1.64	2.32	2.30	3.24	1164.1
10	[4 , 12]	2	2	C	3137	2566	2582	2535	1.49	2.13	2.12	2.98	1029.8
10	[4 , 12]	3	2	C	3075	2478	2494	2448	1.60	2.22	2.28	3.18	1275.9
10	[4 , 12]	4	2	C	2970	2536	2552	2506	1.62	2.31	2.31	3.20	1261.8
10	[4 , 12]	2	3	C	3039	2424	2439	2394	1.61	2.21	2.27	3.20	1218.7
10	[4 , 12]	2	4	C	2904	3308	3329	3268	1.59	2.32	2.30	3.23	1264.9
10	[6 , 16]	2	2	B	3964	3371	3391	3330	2.35	3.53	2.78	4.70	1598.8
10	[6 , 16]	3	2	B	4039	3346	3367	3306	2.32	3.15	2.84	5.21	1699.4
10	[6 , 16]	4	2	B	4010	3167	3187	3129	2.43	3.09	3.41	4.11	1715.3
10	[6 , 16]	2	3	B	3795	3086	3105	3049	2.65	3.22	2.58	3.25	1708.9
10	[6 , 16]	2	4	B	3698	11000	11015	10920	2.26	2.88	2.95	4.05	1769.2
50	[2 , 10]	2	2	A	13245	11080	11102	10992	12.27	17.04	16.96	23.84	*
50	[2 , 10]	3	2	A	13239	11103	11124	11015	12.80	17.75	18.23	24.97	*
50	[2 , 10]	4	2	A	13248	10949	10940	10893	12.96	18.40	18.47	25.59	*
50	[2 , 10]	2	3	A	13028	11082	11080	11018	12.87	17.66	18.15	25.60	*
50	[2 , 10]	2	4	A	13196	11544	11615	11404	13.12	18.55	18.38	25.90	*
50	[4 , 12]	2	2	C	13832	11215	11284	11079	12.68	17.02	17.32	23.82	*
50	[4 , 12]	3	2	C	13438	11488	11559	11349	12.79	17.76	18.22	25.41	*
50	[4 , 12]	4	2	C	13765	11375	11446	11237	12.96	18.47	18.47	25.60	*
50	[4 , 12]	2	3	C	13631	11252	11321	11115	12.87	17.68	18.14	25.57	*
50	[4 , 12]	2	4	C	13483	15448	15543	15260	12.71	18.55	18.39	24.89	*
50	[6 , 16]	2	2	B	18510	15509	15605	15321	18.79	28.23	26.87	37.58	*
50	[6 , 16]	3	2	B	18584	15453	15549	15266	18.57	25.16	27.19	41.68	*
50	[6 , 16]	4	2	B	18517	15223	15316	15038	19.41	26.64	27.25	32.81	*
50	[6 , 16]	2	3	B	18240	15451	15546	15264	21.18	25.71	20.67	25.97	*
50	[6 , 16]	2	4	B	18514	11252	11321	11115	18.06	23.01	23.59	32.42	*
100	[2 , 10]	2	2	A	13483	15448	15543	15260	26.30	36.48	36.32	51.20	*
100	[2 , 10]	3	2	A	18510	15509	15605	15321	27.50	38.06	39.06	53.52	*
100	[2 , 10]	4	2	A	18584	15453	15549	15266	27.86	39.37	39.64	54.88	*
100	[2 , 10]	2	3	A	18517	15223	15316	15038	27.59	37.89	39.00	54.90	*
100	[2 , 10]	2	4	A	18240	15451	15546	15264	28.15	39.75	39.47	55.63	*
100	[4 , 12]	2	2	C	18514	21587	21720	21325	27.19	36.57	37.18	51.00	*
100	[4 , 12]	3	2	C	25866	21637	21770	21374	27.41	38.08	39.08	54.39	*
100	[4 , 12]	4	2	C	25926	22157	22294	21888	27.79	39.54	39.53	54.86	*
100	[4 , 12]	2	3	C	26550	21767	21901	21503	27.58	37.94	39.01	54.84	*
100	[4 , 12]	2	4	C	26082	21775	21970	21571	27.26	39.81	39.51	53.49	*
100	[6 , 16]	2	2	B	26165	23025	23167	22746	40.26	60.52	57.76	73.57	*
100	[6 , 16]	3	2	B	27590	22562	22701	22288	39.79	54.01	58.32	75.57	*
100	[6 , 16]	4	2	B	27034	22539	22678	22265	40.03	57.12	58.51	72.37	*

Table 3. Continued

100	[6 , 16]	2	3	B	27007	22172	22308	21902	41.96	55.17	44.25	69.58	*
100	[6 , 16]	2	4	B	26567	22329	22466	22058	38.68	49.32	50.56	69.59	*
150	[2 , 10]	2	2	A	26755	30720	30909	30347	38.56	53.64	53.25	75.00	*
150	[2 , 10]	3	2	A	36810	30107	30191	29642	40.22	55.78	57.29	78.47	*
150	[2 , 10]	4	2	A	35954	30000	30185	29636	40.72	57.92	58.19	80.55	*
150	[2 , 10]	2	3	A	35948	30000	30105	29715	40.43	55.49	57.09	80.55	*
150	[2 , 10]	2	4	A	36043	30006	30166	29618	41.25	58.40	57.81	81.39	*
150	[4 , 12]	2	2	C	35925	32829	33031	32430	39.87	53.48	54.51	75.04	*
150	[4 , 12]	3	2	C	39337	32755	32957	32358	40.18	55.90	57.32	79.96	*
150	[4 , 12]	4	2	C	39249	32922	33065	32583	40.76	58.07	58.16	80.47	*
150	[4 , 12]	2	3	C	39521	32878	33080	32479	40.48	55.54	57.12	77.46	*
150	[4 , 12]	2	4	C	39396	32722	32923	32324	44.97	58.32	57.77	78.24	*
150	[6 , 16]	2	2	B	39208	33584	33675	33292	59.13	88.72	84.39	98.34	*
150	[6 , 16]	3	2	B	40382	33426	33440	33212	58.41	79.20	85.56	91.15	*
150	[6 , 16]	4	2	B	40285	33392	33449	33135	61.00	83.74	85.68	93.20	*
150	[6 , 16]	2	3	B	40192	33317	33423	33121	60.64	70.84	68.94	81.77	*
150	[6 , 16]	2	4	B	40043	33261	33396	33108	56.75	72.42	70.12	92.08	*
Average					19749.22	17001.45	17090.75	16816.37	23.84433	32.716	32.73033	43.627	*

The effect of the number of product on the objective function is shown in figure 5. These results show that the algorithm *JIV* has had the best performance compared to the other algorithms in four categories of the problems. This figure also presents the algorithm *JI* as the worst one with the significant differences between the results of this algorithm and the rest in all categories.

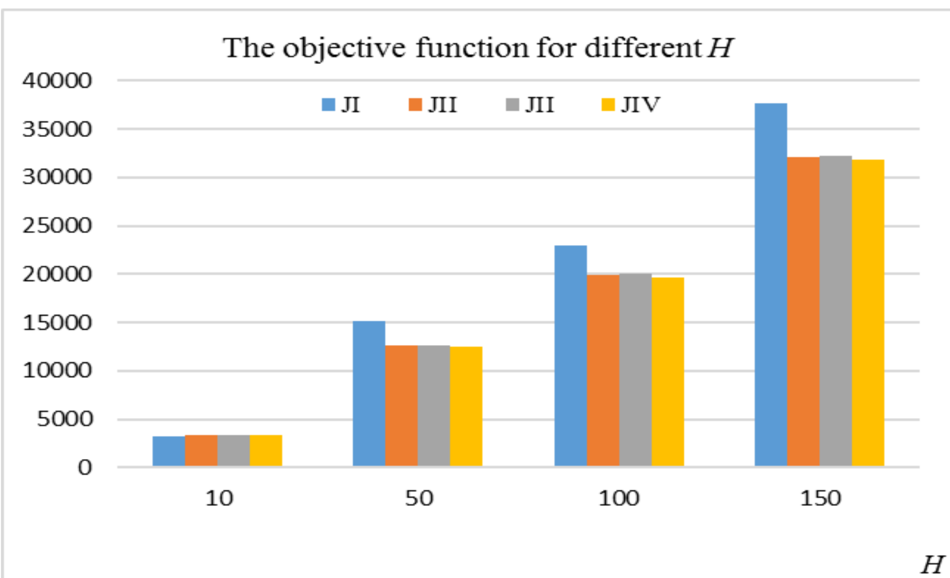


Figure 4. The effect of the number of product on the objective

6. Summary and conclusion

The problem considered in this study is described as follows: several products of different kinds must be produced. Each product needs a set of parts to complete. Firstly the parts are fabricated on a two stage hybrid flow shop in which that there are different number of identical machines in each stage. After the hybrid flow shop, there is an assembly stage where parts are put together into the assembled product. Aging effects of the hybrid flow shop machines and preventive maintenance activities is considered for the fabricator machines. The aim is to determine sequence of products to be assembled, scheduling the parts and assigning them to machines in each stage of hybrid flow shop to be fabricated, and also determine when the preventive maintenance activities get done. The considered objective function was to minimize the completion time of all products (makespan). The parameters and variables of this problem were introduced and then its mathematical model was developed. The developed mathematical model was run with GAMS program for the small scale test problems. Since this problem has been proved strongly NP-hard, in order to solve the problem in medium and large scales a series of heuristic algorithms is proposed based on Johnson's algorithm (*JI JII*, *JIII*, and *JIV*). In order to test the proposed algorithms, three situations were considered: 1. when the stage of part fabrication is bottleneck, 2. when assembly stage is bottleneck and 3. When there is a balanced condition.

The result shows that when hybrid flow shop is bottleneck, the greatest improvement is obtained on the objective function. Also the results show algorithm JIV has the best performance because of the more accurate computations of the Johnson's algorithm. That is because the process time of the parts of each product in hybrid flow shop is more accurate in algorithm *JIV*. On the other hand, the run time of algorithm JI is least of all because its computation is simpler for the Johnson's algorithm.

Studying this problem with a number of similar products may be interesting for further research. Also considering limitation in buffers is suggested for future studies. Adding new objectives to this problem and trying to solve it as a multi objective problem can be useful for research.

References

A. Allahverdi, FS.Al-Anzi. The two-stage assembly scheduling problem to minimize total completion time with setup times. *Computers & Operations Research.* 36:2740-2747, 2009.

A. Bachman, A. Janiak. Scheduling jobs with position-dependent processing times. *Journal of Operational Research Society*. 55, 257-264, 2004.

A. Berrichi, L. Amodeo, F. Yalaoui, E. Chatelet, M. Mezghiche. Bi-objective optimization algorithms for joint production and maintenance scheduling: Application to the parallel machine problem. *Journal of Intelligent Manufacturing*. 20, 389–400, 2008.

AMA. Hariri, CN. Potts, A branch and bound algorithm for the two-stage assembly scheduling problem. *European Journal of Operational Research.* 103:547-556, 1997.

B. Alidaee, N.K. Womer. Scheduling with time dependent processing times: review and extensions. *Journal of the Operational Research Society*. 50: 711-720, 1999.

C. Oguz, MF. Ercan, TCE. Cheng, YF. Fung. Heuristic algorithms for multiprocessor task scheduling in a two-stage hybrid flow-shop. *European Journal of Operational Research.* 149:390-403, 2003.

C. Sun, K. Morizawa, H. Nagasawa.Powerful heuristics to minimize makespan in fixed, 3-machine, assembly-type flow shop scheduling. *European Journal of Operational Research.* 146:498-516, 2003

Ch. Koulamas, GJ. Kyparisis. The three-stage assembly flow shop scheduling problem. *Computers & Operations Research* 28:689-704, 2001.

CN. Potts, SV. Sevast'Janov, VA. Strusevich, LN. Van Wassenhove, CM. Zwaneveld. The two-stage assembly scheduling problem: Complexity and approximation. *Operations Research.* 43:346-355, 1995.

CY. Lee, TCE. Cheng, BMT. Lin. Minimizing the makespan in the 3-machine assembly-type flowshop scheduling problem. *Management Science* 39:616-25, 1993.

D. Quadt, H. Kuhn. A taxonomy of flexible flow line scheduling procedures. *European Journal of Operational Research.* 178:686-698, 2007.

D. Yang, TCE. Cheng, S. Yang, Ch. Hsu. Unrelated parallel-machine scheduling with aging effects and multi-maintenance activities. *Computers & Operations Research*. 39, 1458-1464, 2012.

FS. Al-Anzi, A. Allahverdi, Heuristics for a two-stage assembly flow shop with bicriteria of maximum lateness and makespan. *Computers & Operations Research.* 36:2682-2689, 2009.

G. Mosheiov. Parallel machine scheduling with a learning effect. *Operational Research Society*. 52, 1165-1169, 2001.

G. Schmidt. Scheduling with limited machine availability, *European Journal of Operational Research.* 121: 1-15, 2000.

H. Chou-Jung Hsu. Single-Machine Scheduling with Aging Effects and Optional Maintenance Activity Considerations. *Mathematical Problems in Engineering*. DOI: 10.1155/2013/634503, 2013.

H. Mokhtari, M. Dadgar. A Flexible Job Shop Scheduling Problem with Controllable Processing Times to Optimize Total Cost of Delay and Processing. *International Journal of Supply and Operations Management.* 2(3): 871-887, 2015.

J. Blazewicz, K. EckeK, E. Pesch, G. Schmidt, J. Weglarz. Handbook on Scheduling from Theory to Application. Springer, USA, 2007.

M. Elbounjimi, G. Abdul-Nour, D. Ait-Kadi. A collocation-based approach to designing remanufacturing closed–loop supply chain network. *International Journal of Supply and Operations Management.* 2(3): 820-832, 2015.

M. Yokoyama. Hybrid flow-shop scheduling with assembly operations. *International Journal of Production Economics.* 73:103-116, 2001

M. Yokoyama, DL. Santos. Three-stage flow-shop scheduling with assembly operations to minimize the weighted sum of product completion times. *European Journal of Operational Research.* 161:754-770, 2005.

ML. Pinedo. Scheduling Theory, Algorithms, and Systems. Third Edition. Springer, USA, 2008.

TCE. Cheng, G. Wang. Scheduling the fabrication and assembly of components in a two-machine flow shop. *IIE Transactions.* 31:135-143, 1999.

P. Fattahi, SMH. Hosseini, F. Jolai, R. Tavakoli-Moghadam. A branch and bound algorithm for hybrid flow shop scheduling 4 problem with setup time and assembly operations. *Applied Mathematical Modelling*. 38, 119-134, 2014.

P. Fattahi, SMH. Hosseini, F. Jolai. A mathematical model and extension algorithm for assembly flexible flow shop scheduling problem. *International journal of advanced manufacturing technology*. 65:787–802, 2013

R. Ruiz, JA. Vazquez-Rodriguez. Invited Review The hybrid flow shop scheduling problem. *European Journal of Operational Research.* 205:1-18, 2010.

T.C.E. Cheng, O. Ding, B.M.T. Lin. A concise survey of scheduling with time-dependent processing times. *European Journal of Operational Research*. 152: 1-13, 2004.

Y. Ma, C. Chu, C. Zuo A survey of scheduling with deterministic machine availability constraints. *Computers & Industrial Engineeri*