*Proceedings of the 2015 International Conference on Operations Excellence and Service Engineering*
*Orlando, Florida, USA, September 10-11, 2015*

# Multi-Objective Non-Permutation Flowshop with Dependent Setup Times and Missing Operations

**Shaya Sheikh**
Management Science Department
New York Institute of Technology
1855 Broadway, New York, NY 10023
shaya.sheikh@case.edu

**Mohammad Komaki**
Electrical Engineering and Computer Science Department
Case Western Reserve University
Euclid Avenue, Cleveland, USA
gxk152@case.edu

**Ehsan Teymourian**
Department of Industrial Engineering
Mazandaran University of Science and Technology
Babol, Iran
ehsan.teymorian@gmail.com

**Behnam Malakooti**
Electrical Engineering and Computer Science Department
Case Western Reserve University
Euclid Avenue, Cleveland, USA
gxk152@case.edu

## Abstract

In this paper, a mixed-integer linear programming (MILP) model is developed to solve non-permutation flowshop problem with objectives of minimizing makespan, increasing service level, and maximizing job priorities. The presented model can be applied in a variety of industries such as pharmaceutical, steel manufacturing or other continuous or batch process industries. Setup times are assumed to be dependent to sequence of jobs on each stage and jobs are allowed to skip one or more stages. Available operating hours are also imposed to the problem. The presented model is the generalization of model presented by Srikar and Ghosh (1986) and Mehravaran and Logendran (2012). A framework for solving and analyzing the problem is presented. The numerical example shows the general relationship between objectives and gives insights on how the framework can be applied to industrial sized problems.

## Keywords
Non-permutation flowshop, Multi-objective, Sequence-dependent setup times

## 1. Introduction
Flow shop consists of series of stages in which N jobs have to be processed on M stages. In flow shop, every job has to be processed at most once on a stage, and each stage can only process one job at a time. Processing of a job must be completed on the current stage before the job being passed on to the next stage. In non-permutation schedule, jobs can have different sequence on different stages. In this paper, a flowshop scheduling problem with sequence-dependent setup times and a multi-objective objective of minimizing makespan, maximizing service level, and maximizing job

priorities is developed. These objectives are considered among the most important objectives in wide ranges of production and manufacturing factories. A mixed integer linear mathematical model for solving the non-permutation flowshop is developed to address all of the operational constraints commonly encountered in the industry, including the possibility of jobs skipping one or more stages and considering operating hours.

Non-permutation flow shops with sequence dependent setup times and possibility to skip stages is observed frequently in many manufacturing or production fields including pharmaceutical companies, steel production companies, electronics manufacturing, space shuttle processing, or food producing companies. In these facilities, in-process ingredients may skip stages depending on the recipe or desired final product grade. Sequence-dependency of setup times implies that the setup time required of a job on a stage is dependent on the previously processed job on that stage. We assume that all jobs and stages are available at the beginning of the scheduling process. Stage skipping is addressed by a binary variable to represent processing routes.

Researchers have developed heuristic algorithms and optimization models to solve non-permutation flow shop problem effectively. Rossi and Lanzetta (2014) presented meta-heuristic approach for improving existing solutions for general non-permutation flow shops. Among other recent researches, we can refer to Aggoune & Portmann (2006), Chang, Chen, & Liu (2007), Liao & Huang (2010), Mehravaran & Logendran (2012), Ruiz & Maroto (2005), Komaki et al. (2014a,b,c), Yagmahan & Yenisey (2010), Ying, (2008), and Kuo et al. (2009). Eren and Guner (2006) developed a mixed-integer programming model to find the optimum schedule for a single-stage problem with sequence-dependent setup times. Liao and Huang (2010) developed a non-permutation schedule for minimizing the tardiness. Ying (2008) constructed a greedy heuristic for solving non-permutation flowshop with the goal of minimizing the makespan. Multi-objective optimization of scheduling problems has been discussed by many researchers including Eren and Guner (2006), Mansouri et al. (2009), Koksalan and Burak Keha (2003), Chen & Vairaktarakis (2005), Moslehi et al. (2009), Mehravaran (2001), and Choua and Lee (1999).

The presented model is the generalization of model presented by Srikar and Ghosh (1986) and Mehravaran and Logendran (2012). In this paper, a multi-objective optimization is presented in order to minimize the makespan, to increase service level, and to maximize job priorities. Note that increasing service level is translated into minimizing tardiness. There is a weight assigned to each objective that shows the relative importance of objectives for decision maker.

The main contributions of this paper is summarized as:

1. Considering constraints such as sequence-dependent setup time, operation hours, stage skipping, with multi-objective optimization.
2. Suggesting a framework for solving and analyzing the presented problem.

The rest of this paper is organized as follows. In Section 2, elements for developing suggested non-permutation flowshop scheduling problem is introduced. Section 3 formulates the problem. Experimental results are shown in Section 4. Section 5, concludes and presents the directions for future research.

## 2. Model Definition

(Garey et al., 1976) proved that two-stage flow shop scheduling problem with minimization of sum of weighted completion times is strongly NP-hard. Single-stage scheduling problem with minimization of sum of weighted tardiness had also been proven to be strongly NP-hard (Lenstra et al., 1977). Developed model in this paper with search space of $(N!)^M$ sequences, is a generalization of aforementioned models. Therefore the presented problem is also strongly NP-hard.

Maximizing service level and job priorities can be translated into minimizing tardiness and minimizing the deviation from preferred producer’s sequence, respectively. Defined parameters and decision variables are presented in nomenclature.

**Nomenclature**

| **Index** | **Decision Variables** |
|---|---|
| j: Job j where j=1,…, N<br>i: Stage i where i=1,…, M<br>**Parameters**<br>$t_{i,j}$: Processing time of job j on stage i<br>$s_{i,k,j}$: Setup times from job k to job j on stage i<br>$r_{i,j}$ : Processing route for job j on stage i. $r_{i,j}$ is 1 if job j needs to be processed on stage i and 0 otherwise.<br>Δ: A fixed large number<br>$g_j$: Goal sequence of jobs on last stage<br>$w_1$, $w_2$, $w_3$: Weights of importance for objectives $f_1$, $f_2$, and $f_3$<br>$d_j$ : Due date for job j | $C_{i,j}$: Completion time of job j on stage i.<br>Makespan (or $f_1$): Time to finish processing of all jobs on the last stage.<br>$P_{i,k,j}$: Precedence of job j with respect to job k in stage i. Precedence is 1 if job k completes before job j starts and 0 otherwise.<br>$o_j$: Current sequence (optimal sequence without considering producer's priority) of jobs on last stage<br>$\bar{T} = \frac{1}{N}\sum_{J=1}^{N} \text{Max}\,[0, (C_{M,j} - d_j)]$ (or $f_2$): Average Tardiness<br>$\sum_{j=1}^{n} \mid o_j - g_j \mid$ (or $f_3$): Difference between the goal sequence and the current sequence |

Tardiness shows whether a job has passed its due date. In average tardiness formula, $\bar{T} = \frac{1}{N}\sum_{J=1}^{N} \text{Max}\,[0, (C_{M,j} - d_j)]$, $\bar{T}$ represents average tardiness of all jobs, $d_j$ is the due date for job j, and $C_{M,j}$ shows the completion time of job j in the last stage.

**Bi-Objective Analysis:** Consider the following two objectives

Minimize $f_1$ = Makespan

Minimize $f_2$ = Average Tardiness (or $\bar{T}$)

The first objective also minimizes the average waiting time for jobs in the flow shop and the second objective may also minimize the average lateness. For a given additive utility function, the following method can be used to generate the best alternative.

Minimize $U = w_1 * f_1 + w_2 * f_2$ where $w_1 + w_2 = 1$

where $w_1$ and $w_2$ are the weights of importance for $f_1$ and $f_2$, respectively. The approach for generating efficient alternatives for bi-objective sequencing problems is similar to the approach of multi-objective problem which will be explained below.

**Multi-Objective Analysis:** Tri-objective sequencing problems is formulated as

$f_1$ = Minimize Makespan

$f_2$ = Minimize the average tardiness ($\bar{T}$)

$f_3$ = Maximize job priorities (producer's priority)

$f_1$ and $f_2$ are measured by makespan and $\bar{T}$ respectively. $f_3$ is measured through following approach.

Objective, $f_3$, minimize the difference between the current sequence of jobs on last stage and the producer's priority or goal sequence of jobs on last stage. Sequence of jobs on last stage implies how soon a product is ready for delivery to customer.

$$\text{Minimize } f_3 = \sum_{j=1}^{n} \mid o_j - g_j \mid$$

Where $o_j$ is the position (or rank) of job j in the current sequence, and $g_j$ is the position of job j in the goal sequence. If the current sequence is the same as the goal sequence, then, $f_3 = 0$; this is the best possible solution for job priorities. For given weights of importance for the three objectives, $f_1$, $f_2$, and $f_3$, we use the following step by step approach to find the optimal solution:

1. Find the sequence that minimizes makespan by minimizing Equation (6) in Multi-Objective Flowshop model (or MOFS model) presented in Section 3 subject to Equations (9-16). Then, find corresponding values for $f_2$ and $f_3$.

2. Find the sequence that minimizes average tardiness by minimizing Equation (7) in MOFS model subject to

Equations (9-16). Then, find corresponding values for $f_1$ and $f_3$.

3. Find the sequence that minimizes job by minimizing Equation (8) in MOFS model subject to Equations (9-16). Then, find corresponding values for $f_1$ and $f_2$.

Following table can be constructed using solution to the above problems

| Problems | Solution | | |
|---|---|---|---|
| | $f_1$ | $f_2$ | $f_3$ |
| Problem 1: Minimize $f_1$ S.T. Constraints | $f_{1,1}$ | $f_{2,1}$ | $f_{3,1}$ |
| Problem 2: Minimize $f_2$ S.T. Constraints | $f_{1,2}$ | $f_{2,2}$ | $f_{3,2}$ |
| Problem 3: Minimize $f_3$ S.T. Constraints | $f_{1,3}$ | $f_{2,3}$ | $f_{3,3}$ |
| Maximum | $f_{1,max}$ | $f_{2,max}$ | $f_{3,max}$ |
| Minimum | $f_{1,min}$ | $f_{2,min}$ | $f_{3,min}$ |

4. Solve MOFS model (all equations) by incorporating maximum and minimum values of the three functions, found in above table, into Equations (3-5).

## 3. Model Formulation

**Multi-Objective Flow-Shop Model (MOFS model)**

$$\text{Minimize } w_1 * f_1' + w_2 * f_2' + w_3 * f_3' \quad (1)$$

$$w_1 + w_2 + w_3 = 1 \quad (2)$$

$$f_1' = (f_1 - f_{1,min}) / (f_{1,max} - f_{1,min}) \quad (3)$$

$$f_2' = (f_2 - f_{2,min}) / (f_{2,max} - f_{2,min}) \quad (4)$$

$$f_3' = (f_3 - f_{3,min}) / (f_{3,max} - f_{3,min}) \quad (5)$$

Subject to:

$$f_1 = \text{Makespan} \quad (6)$$

$$f_2 = \frac{1}{N}\sum_{J=1}^{N} \text{Max}\,[0, (C_{M,j} - d_j)] \quad (7)$$

$$f_3 = \sum_{j=1}^{n} |o_j - g_j| \quad (8)$$

$$C_{i,j} \geq r_{i,j} * t_{i,j} \qquad \text{for } i=1,\ldots,M \text{ and } j=1,\ldots,N \quad (9)$$

$$C_{i+1,j} \geq C_{i,j} + t_{i+1,j} * r_{i+1,j} \qquad \text{for } i=1,\ldots,M\text{-}1 \text{ and } j=1,\ldots,N \quad (10)$$

$$r_{i,j}*(C_{i,j} - C_{i,k}) + \Delta * P_{i,k,j} \geq r_{i,j}*(t_{i,j} + s_{i,k,j}) \qquad \text{for } i=1,\ldots,M, \text{ and } j, k=1,\ldots,N \text{ where } j>k \quad (11)$$

$$r_{i,k}*(C_{i,k} - C_{i,j}) + \Delta*(1 - P_{i,k,j}) \geq r_{i,k}*(t_{i,k} + s_{i,k,j}) \qquad \text{for } i=1,\ldots,M, \text{ and } j, k=1,\ldots,N \text{ where } j>k \quad (12)$$

$$\text{Makespan} \geq C_{M,j} \qquad j=1,\ldots,n \quad (13)$$

$$C_{i,j} \in \text{Operating Hours} \qquad \text{for } i=1,2, M, \text{ and } j, k=1,\ldots,N \quad (14)$$

$$(C_{i,j} - t_{ij}) \in \text{Operating Hours} \quad (15)$$

$$C_{i,j} \text{ and Makespan} \geq 0, \quad P_{i,k,j} \text{ binary variable} \qquad f_1, f_2 \text{ are unrestricted in sign} \quad (16)$$

Utility function (1) focuses on minimizing the weighted sum of normalized values for makespan, weighted tardiness, and weighted deviation from preferred producer's sequence. Equation (2) enforces summation of all weights to be equal to 1. Equations (3-5) finds the normalized values for objectives where ($f_{1,min}$, $f_{1,max}$), ($f_{2,min}$, $f_{2,max}$), and ($f_{3,min}$, $f_{3,max}$) show minimum and maximum possible value for objective functions. Constrains (6-8) defines objective functions, respectively. Constraints (9) ensure that all jobs are scheduled and completion times of jobs are at least as large as processing time of that job on a particular stage. Constraints (10) relate completion time of a job in one stage to completion time of the same job in previous stage. Constraints (11) and (12) enforces sequence constraints for all jobs on each stage. Constraint (13) finds completion time of the jobs in last stage. Constraints (14) ensures that no job completes in after hours or weekends. Constraint (15) contributes in problem size reduction. Specifically, it allows the stages to start processing jobs in working hours. It ensures that no job starts either during after-hours or weekends. Constraints (16) define the non-negativity requirements on variables $C_{i,j}$ and Makespan, while $P_{i,k,j}$ need to be binary variables.

## 4. Experimental Results

For testing the performance, we solved a flowshop problem with 6 jobs, 4 stages. Specific processing routes, processing times and set-up times are explained in Tables 1 and 2. The platform used is Intel Xeon CPU 3.00 GHz with 4 GB RAM. It is assumed that stages are available 27/4. Therefore, weekend and after-hour constraints were not considered in this example. It is also assumed that producer's preferred sequence of jobs on the last stage is 2-3-1-6-5-4. In the first step, MOFS problem is solved using single objective in order to find maximum and minimum for all three objectives. The results are shown in Table 3.

**Table 1**: Processing time, processing routes, and due dates; t represents processing time, r represent processing route, where r is 1 if job i needs to be processed on stage j and 0 otherwise. d represent due dates

| | $M_1$ | | $M_2$ | | $M_3$ | | $M_4$ | | |
|---|---|---|---|---|---|---|---|---|---|
| Job j' / Job j | $t_{1,j}$ | $r_{1,j}$ | $t_{2,j}$ | $r_{2,j}$ | $t_{3,j}$ | $r_{3,j}$ | $t_{4,j}$ | $r_{4,j}$ | $d_j$ |
| **1** | 7 | 1 | 5 | 1 | 6 | 1 | 8 | 1 | 220 |
| **2** | 6 | 1 | 9 | 0 | 1 | 1 | 4 | 1 | 210 |
| **3** | 4 | 0 | 6 | 1 | 5 | 1 | 1 | 1 | 200 |
| **4** | 5 | 1 | 5 | 1 | 5 | 1 | 3 | 1 | 200 |
| **5** | 7 | 1 | 3 | 0 | 4 | 1 | 6 | 1 | 200 |
| **6** | 2 | 1 | 3 | 1 | 2 | 1 | 3 | 0 | 190 |

**Table 2**: Setup times from job k to job j on stage i

| | $M_1$ | | | | | | $M_2$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Job j / Job k | **1** | **2** | **3** | **4** | **5** | **6** | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | 2 | 0 | 3 | 4 | 5 | 5 | 3 | 6 | 2 | 1 | 1 | 5 |
| **2** | 6 | 1 | 2 | 3 | 5 | 4 | 4 | 2 | 2 | 1 | 5 | 6 |
| **3** | 4 | 0 | 2 | 1 | 8 | 5 | 1 | 2 | 3 | 4 | 5 | 6 |
| **4** | 5 | 1 | 2 | 3 | 5 | 7 | 3 | 5 | 4 | 1 | 2 | 2 |
| **5** | 1 | 0 | 2 | 3 | 6 | 5 | 6 | 5 | 4 | 3 | 2 | 1 |
| **6** | 2 | 1 | 3 | 4 | 8 | 6 | 3 | 6 | 2 | 4 | 1 | 5 |

| | $M_3$ | | | | | | $M_4$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Job j / Job k | **1** | **2** | **3** | **4** | **5** | **6** | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | 6 | 1 | 2 | 3 | 5 | 4 | 4 | 2 | 2 | 1 | 5 | 6 |
| **2** | 4 | 0 | 2 | 1 | 8 | 5 | 1 | 2 | 3 | 4 | 5 | 6 |
| **3** | 3 | 4 | 2 | 3 | 8 | 5 | 5 | 5 | 4 | 3 | 2 | 2 |
| **4** | 1 | 1 | 3 | 4 | 5 | 8 | 3 | 6 | 2 | 4 | 5 | 5 |
| **5** | 5 | 1 | 2 | 3 | 5 | 7 | 3 | 5 | 4 | 1 | 2 | 2 |
| **6** | 1 | 0 | 2 | 3 | 6 | 5 | 6 | 5 | 4 | 3 | 2 | 1 |

Table 3: Solution of Problems I, II, III

| Problems | Solution | | |
|---|---|---|---|
| | $f_1$ | $f_2$ | $f_3$ |
| Problem I: Minimize $f_1$ S.T. Constraints of MOFS problem | 198 | 15 | 3 |
| Problem II: Minimize $f_2$ S.T. Constraints of MOFS problem | 221 | 2 | 2 |
| Problem III:Minimize $f_3$ S.T. Constraints of MOFS problem | 220 | 18 | 0 |
| Maximum | 221 | 18 | 3 |
| Minimum | 198 | 2 | 0 |

Table 4: Solutions for multi-objective example

| **Weights** | | | **Objectives** | | | **Normalized Objectives** | | | **Normalized Utility** |
|---|---|---|---|---|---|---|---|---|---|
| $w_1$ | $w_2$ | $w_3$ | Min $f_1$ | Min $f_2$ | Min $f_3$ | $f_1'$ | $f_2'$ | $f_3'$ | |
| 0.1 | 0.1 | 0.8 | 220 | 15 | 0 | 211.4 | 1.0 | 0.0 | 21.2 |
| 0.1 | 0.2 | 0.7 | 220 | 15 | 1 | 211.4 | 0.8 | 0.3 | 21.5 |
| 0.1 | 0.3 | 0.6 | 219 | 15 | 1 | 210.4 | 0.8 | 0.3 | 21.5 |
| 0.1 | 0.4 | 0.5 | 220 | 10 | 1 | 211.4 | 0.5 | 0.3 | 21.5 |
| 0.1 | 0.5 | 0.4 | 220 | 6 | 1 | 211.4 | 0.3 | 0.3 | 21.4 |
| 0.1 | 0.6 | 0.3 | 220 | 6 | 2 | 211.4 | 0.3 | 0.7 | 21.5 |
| 0.1 | 0.7 | 0.2 | 220 | 6 | 2 | 211.4 | 0.3 | 0.7 | 21.4 |
| 0.1 | 0.8 | 0.1 | 221 | 2 | 2 | 212.4 | 0.0 | 0.7 | 21.3 |
| 0.2 | 0.1 | 0.7 | 218 | 15 | 0 | 209.4 | 1.0 | 0.0 | 42.0 |
| 0.2 | 0.2 | 0.6 | 216 | 15 | 1 | 207.4 | 0.8 | 0.3 | 41.8 |
| 0.2 | 0.3 | 0.5 | 218 | 15 | 1 | 209.4 | 0.8 | 0.3 | 42.3 |
| 0.2 | 0.4 | 0.4 | 218 | 15 | 1 | 209.4 | 0.8 | 0.3 | 42.3 |
| 0.2 | 0.5 | 0.3 | 216 | 10 | 1 | 207.4 | 0.5 | 0.3 | 41.8 |
| 0.2 | 0.6 | 0.2 | 215 | 10 | 2 | 206.4 | 0.5 | 0.7 | 41.7 |
| 0.2 | 0.7 | 0.1 | 216 | 6 | 3 | 207.4 | 0.3 | 1.0 | 41.8 |
| 0.3 | 0.1 | 0.6 | 210 | 15 | 1 | 201.4 | 0.8 | 0.3 | 60.7 |
| 0.3 | 0.2 | 0.5 | 210 | 15 | 1 | 201.4 | 0.8 | 0.3 | 60.7 |
| 0.3 | 0.3 | 0.4 | 210 | 15 | 1 | 201.4 | 0.8 | 0.3 | 60.8 |
| 0.3 | 0.4 | 0.3 | 209 | 15 | 2 | 200.4 | 0.8 | 0.7 | 60.6 |
| 0.3 | 0.5 | 0.2 | 209 | 9 | 2 | 200.4 | 0.4 | 0.7 | 60.5 |
| 0.3 | 0.6 | 0.1 | 209 | 6 | 2 | 200.4 | 0.3 | 0.7 | 60.3 |
| 0.4 | 0.1 | 0.5 | 206 | 15 | 1 | 197.4 | 0.8 | 0.3 | 79.2 |
| 0.4 | 0.2 | 0.4 | 206 | 15 | 1 | 197.4 | 0.8 | 0.3 | 79.3 |
| 0.4 | 0.3 | 0.3 | 206 | 15 | 1 | 197.4 | 0.8 | 0.3 | 79.3 |
| 0.4 | 0.4 | 0.2 | 206 | 6 | 2 | 197.4 | 0.3 | 0.7 | 79.2 |
| 0.4 | 0.5 | 0.1 | 205 | 9 | 2 | 196.4 | 0.4 | 0.7 | 78.8 |
| 0.5 | 0.1 | 0.4 | 205 | 15 | 1 | 196.4 | 0.8 | 0.3 | 98.4 |
| 0.5 | 0.2 | 0.3 | 206 | 9 | 2 | 197.4 | 0.4 | 0.7 | 99.0 |
| 0.5 | 0.3 | 0.2 | 205 | 9 | 2 | 196.4 | 0.4 | 0.7 | 98.5 |
| 0.5 | 0.4 | 0.1 | 201 | 15 | 2 | 192.4 | 0.4 | 0.7 | 96.4 |
| 0.6 | 0.1 | 0.3 | 201 | 15 | 1 | 192.4 | 0.8 | 0.3 | 115.6 |
| 0.6 | 0.2 | 0.2 | 200 | 15 | 2 | 191.4 | 0.8 | 0.7 | 115.1 |
| 0.6 | 0.3 | 0.1 | 200 | 15 | 2 | 191.4 | 0.4 | 0.7 | 115.0 |
| 0.7 | 0.1 | 0.2 | 200 | 15 | 2 | 191.4 | 0.8 | 0.7 | 134.2 |
| 0.7 | 0.2 | 0.1 | 200 | 15 | 3 | 191.4 | 0.8 | 1.0 | 134.2 |
| 0.8 | 0.1 | 0.1 | 198 | 18 | 3 | 189.4 | 0.8 | 1.0 | 151.7 |

Table 4 shows experimental results for multi-objective example. It also explains the relationship between makespan, tardiness, and producer's preferred sequence. Corresponding weights for these objectives are changed in a range from 0.1 to 0.8. If one of the weights becomes zero, then the problem simply becomes the biobjective optimization problem. In Table 4, a total number of 36 weight combinations are used to generate solutions. The values in these figures are normalized in order to simplify its analysis. A general indirect relationship between tardiness ($f_2$) and producer's preferred sequence ($f_3$) and between makespan ($f_1$) and ($f_3$) can be detected.

## 5. Conclusions and Future Research

In this paper, a non-permutation flowshop with sequence-dependent setup times and a framework for solving it was developed. Optimal value for three objectives of makespan, tardiness, and Problems were found by varying weights for corresponding objective functions. Generated solutions show the relationships between these objectives. Results suggests that there is conflicting relationship between makespan and tardiness, and between makespan and producer's preferred sequence. Presented model has the potential to substantially reduce the makespan and tardiness while the generated result being as much close as possible to producer's preferred sequence. In the future, we plan to explore other factors such as throughput rate, waste rate, downtime frequency, work in-process inventory between stages, no waiting time condition, and capacity constraints on different stages.

## References


Aggoune, R., & Portmann, M. C. (2006). Flow shop scheduling problem with limited stage availability: A heuristic approach. International Journal of Production Economics, 99(1–2), 4–15.

Chang, P. C., Chen, S. H., & Liu, C. H. (2007). Sub-population genetic algorithm with mining gene structures for multiobjective flowshop scheduling problems. Expert systems with applications, 33(3), 762–771.

Chen, Z., & Vairaktarakis, G. (2005). Integrated scheduling of production and distribution operations. Management Science, 51(4), 614–628.

Eren, T., Guner,¨ E., 2006. A bicriteria scheduling with sequence-dependent setup times. Applied Mathematics and Computation 179 (1), 378–385.

Garey, M. R., Johnson, D. S., & Sethi, R. (1976). The complexity of flowshop and jobshop scheduling. Mathematics and Operations Research, 1(2), 117–129.

ILOG CPLEX Optimization Studio. (2015). IBM, Version 12.6.

Koksalan, M., Burak Keha, A., 2003. Using genetic algorithms for single-stage bicriteria scheduling problems. European Journal of Operational Research 145 (3), 543–556.

Komaki, M., S. Sheikh, E. Teymourian, 2014a"Hybrid Harmony Search Algorithm to minimize total weighted tardiness in permutation flow shop", *IEEE Symposium Series on Computational Intelligence*, Orlando, Florida.

Komaki, M., S. Sheikh, B. Malakooti, 2014b "Ruled Based Approach for Scheduling Flow-Shop and Job-shop Problems." *ICAM-2014*, Rochester, MI, 28-30 May, 2014.

Komaki, M., S. Sheikh, B. Malakooti, 2014c "Multi-Objective Scheduling Using Rule Based Approach." *ICAM-2014*, Rochester, MI, 28-30, May 2014.

Kuo, I., Horng, S. J., Kao, T. W., Lin, T. L., Lee, C. L., Terano, T., et al. (2009). An efficient flow-shop scheduling algorithm based on a hybrid particle swarm optimization model. Expert systems with applications, 36(3), 7027–7032.

Lenstra, J. K., Rinnooy Kan, A. H. G., & Brucker, P. (1977). Complexity of stage scheduling problems. Annals of Discrete Mathematics, 343–362.

Liao, L.M., Huang, C.J., 2010. Tabu search for non-permutation flowshop scheduling problem with minimizing total tardiness. Applied Mathematics and Computa-tion 217 (2), 557–567.

Logendran, R., McDonell, B., Smucker, B., 2007. Scheduling unrelated parallel stage with sequence-dependent setups. Computers and Operations Research 34 (11), 3420–3438.

Malakooti, B. (2014). Operations and Productions Systems with Multi-Objectives, John Wiley.

Mansouri, A., Hendizadeh, H., & Salmasi, N. (2009). Bicriteria scheduling of a twostage flowshop with sequence-dependent setup times. International Journal of Advance Manufacturing Technology, 40(11), 1216–1226.

Mehravaran, Y., & Logendran, R. (2011). Bicriteria supply chain scheduling on unrelated-parallel stages. Journal of the Chinese Institute of Industrial Engineers, 28(2), 91–101.

Mehravaran Y., R. Logendran (2012). "Non-permutation flowshop scheduling in a supply chain with sequence-dependent setup times."*International Journal of Production Economics*, Vol. 135, Issue 2, 953-963.

Moslehi, G., Mirzaee, M., Vasei, M., Modarres, M., Azaron, A., (2009). Two-stage flow shop scheduling to minimize the sum of maximum earliness and tardiness. International Journal of Production Economics 122 (2), 763–773.

Onwubolu, G., Davendra, D., (2006). Scheduling flow shops using differential evolution algorithm. European Journal

of Operational Research 171 (2), 674–692.
Rossi, A., and M. Lanzetta. "Native metaheuristics for non-permutation flowshop scheduling." *Journal of Intelligent Manufacturing* 25.6 (2014): 1221-1233.
Ruiz, R., Maroto, C., (2005). A comprehensive review and evaluation of permutation flowshop heuristics. European Journal of Operational Research 165 (2), 479–494.
Silva, J. D. L., Burke, E. K., & Petrovic, S. (2004). An introduction to multiobjective metaheuristics for scheduling and timetabling. Lecture Notes in Economics and Mathematical Systems, 535, 91–132.
Srikar, B.N., Ghosh, S., (1986). A MILP model for the n-job, m-stage flowshop with sequence dependent setup times. International Journal of Production Research 24 (6), 1459–1474.
Yagmahan, B., & Yenisey, M. M. (2010). A multi-objective ant colony system algorithm for flow shop scheduling problem. Expert Systems with Applications, 37(2), 1361–1368.
Ying, K. C. (2008). Solving non-permutation flowshop scheduling problems by an effective iterated greedy heuristic. International Journal of Advanced Manufacturing Technology, 38(3), 348–354.

**Biography**

**Shaya Sheikh** obtained his Ph.D. from Case Western Reserve University in 2013. He has worked as Scheduling and Optimization Scientist at Lancaster Laboratories. He is now assistant professor of management science at New York Institute of Technology. His research interests include scheduling, multi-criteria decision making, and energy supply chain management.

**Mohammad Komaki** is accomplished B.Sc. in Industrial Engineering at Sharif University of Technology, Tehran, Iran (2001–2006) and M.Sc. in Industrial Engineering at Mazandaram University of Science & Technology, Babol, Iran (2007–2010). He is currently studying System Engineering at Case Western Reserve University, Cleveland, USA. His research interests include soft computing, optimization and multi-criteria decision making.

**Behnam Malakooti:** Professor Malakooti obtained his PhD in 1982 from Purdue University. He has consulted for numerous industries and corporations, including General Electric, Parker Hannifin, and B.F. Goodrich. He has published over 100 papers in technical journals. In his work, systems architectures, space networks, optimization, multiple criteria & intelligent decision making, trait analysis of biological systems, adaptive artificial neural networks, and artificial intelligence theories and techniques are developed and applied to solve a variety of problems. His current research is on design and protocols analysis for NASA space-based networks. Recently Professor Malakooti developed a four-dimensional approach for decision-making process typology and risk analysis. Decision-making typology accurately identifies the four types of decision makers' behavior: Information processing, creativity, risk, and decisiveness approach. It also provides a basis for developing the next generation of intelligent robots. See http://car.cwru.edu/decision/ for computerized survey. He has made contributions to manufacturing systems developing computer aided approaches for manufacturing/production design, planning, operations, facility layout, assembly systems, scheduling, MEMS, and machine set-up, tool design, and machinability.

**Ehsan Teymourian** received his B.Sc. and M.Sc. in industrial Engineering (IE) from the Mazandaram University of Science & Technology, Babol, Iran (2002-2007–2010). He is now an IE PhD student at MIME department of Oregon State University (OSU), Corvallis, USA. His research interests include Scheduling/Operation Management, Optimization, Production Planning, and Logistics/SCM. He has also published several research papers in prestigious journals and conference proceedings.