**Kamran FORGHANI, PhD candidate**
**Email: kamran21f@gmail.com**
**Department of Industrial Engineering**
**Amirkabir University of Technology, Tehran,Iran**
**Professor Seyyed Mohammad Taghi FATEMI-GHOMI, PhD**
**(Corresponding author)**
**Email: fatemi@aut.ac.i  rand  fatemighomi@gmail.com**
**Department of Industrial Engineering,**
**Amirkabir University of Technology, Tehran, Iran**

## CELLULAR MANUFACTURING SCHEDULING IN THE PRESENCE OF MULTIPLE PROCESS ROUTINGS AND CONSIDERING JOB SPLITTING

*Abstract. The main contribution of this study is the integration of the cell formation, cell scheduling and routing selection problems while considering job splitting and reentrant parts. To the best of our knowledge, this is the first study that investigates such an integrated problem with the job splitting approach. The aim is to find the assignment of machines to the cells, the assignment of operations to the machines, the amount of workload that is dedicated to each machine, and the sequence of operations on machines. The problem is formulated as a mixed-integer program so as to minimize the sum of processing, setup and inter-cell handling costs, plus the total weighted flow time of parts. Due to the computational complexity of the problem, two heuristics are presented to find efficient solutions. Finally, computational experiments are conducted to access the performance of the heuristics.*

*Keywords: Cellular manufacturing system, Cell formation, Scheduling, Job splitting, Multiple process routings, Heuristic.*

**JEL Classification: C44, C52, C61, D24, L60, M11**

## 1. Introduction

Cellular Manufacturing System (CMS) as an approach to implementing the group technology concept, tries to decompose the production system into a set of manufacturing cells so as to facilitate its management. Manufacturing cells are configured by a process called Cell Formation (CF) by which parts are grouped into part families and assigned to machines, and machines are also grouped into a set of cells. The objective is to keep the cells as independent as possible so that one or multiple criteria are met. The main benefits that can be expected from

production by CMSs involve less material handling cost, less setup cost, less production cost, reduction in throughput time, reduction in work-in-process, and easier production control (Singh 1993; Boutsinas 2013).

The majority of early studies on CMSs only focused on the CF problem. However, later, researchers investigated the CF, separately or concurrently with the other important issues in CMSs, like facility layout, scheduling, production planning and reliability. Among these, the scheduling is one of the most important problems that its effective implementation is a requisite for attaining the potential benefits of CMS (Solimanpur and Elmi 2013). If disaggregated manufacturing cells are achieved after the CF process, i.e., parts belonging to the same group (part family)are completely processed in one cell, the resultant scheduling problem is called Group Scheduling (GS) problem. The objective is to determine the sequence of parts within each group, and also the sequence of groups themselves, in a way that some measures of performance are optimized. In this context, the commonly addressed performance criteria for optimization involve the makespan, total weighted tardiness, and total weighted flow time (Wang, Tang, and Yung 2010).

Although perfect disaggregation is ideal in CMSs, however, due to the limitation in resources (machine capacity and budget) or production policies, in all CMS environments it is not possible to achieve such independence (Wemmerlöv and Hyer 1989). As a result, some parts may be processed in more than one cell that in turn causes inter-cell transfers. In the literature, such parts are called as Exceptional Elements (EEs) or exceptional parts. According to Solimanpur and Elmi (2013), the GS in the presence of EEs is called Cell Scheduling (CS). In this respect, Yang and Liao (1996) developed a B&B algorithm, as well as a heuristic, to solve a CS problem which consists of two cells. It was assumed that each job can have at most one operation in each cell. The objective was to minimize the total flow time of jobs. Solimanpur et al. (2004) developed a two-stage heuristic, called SVS-algorithm to solve a CS problem which can contain more than two cells. It was assumed that parts belonging to a part family require the same setup time on each machine, and the setup times are sequence-independent. The objective was to minimize the makespan. Later, in a study conducted by Solimanpur and Elmi (2013), the same problem was solved by a Tabu Search (TS) meta-heuristic. Tavakkoli-Moghaddam et al. (2010) incorporated several objectives into a CS problem by using a weighting sum method for unifying them and solved it by a Scatter Search (SS). The objectives were the minimization of costs arising from makespan, intracellular movements, tardiness, and setups. Zeng et al. (2014) presented a mathematical model for the CS problem in which it was assumed that parts are transported between the cells via an automated guided vehicle. A two-stage GA-based heuristic aiming at minimizing the makespan was used to solve the problem. By considering intercellular transportation times and sequence-dependent family setup times, Halat and Bashirzadeh (2015) formulated a mathematical model for the CS problem and solved it via a Genetic Algorithm (GA). The objective was the minimization of the makespan.

Cellular Manufacturing Scheduling in the Presence of Multiple Process Routings
and Considering Job Splitting
_____

From the perspective of manufacturing, an alternative set of machines may exist for processing each operation. This allows parts to be produced through different process routings. By considering this issue, a decision needs to be made towards the selection of the process routing of parts, which is usually known as Routing Selection (RS) problem. To address this aspect, Kesen and Güngör (2012) formulated a mathematical model for the CS in a virtual CMS environment in which the traveling times were also taken into consideration. They assumed that jobs are produced in lots and the objective is the minimization of the makespan. They solved the problem by using a GA. It should be mentioned that, unlike the typical CMS, in the virtual one, cells are not identifiable as physical groups of machines, but as data files and processes in a computer controller instead. Arkat and Ghahve (2014) presented a bi-objective mathematical model for operations scheduling in a virtual CMS by considering multiple routings, subcontracting and sequence-dependent setup times. The objective functions were the minimization of the total handling costs and the minimization of the makespan. To find a set of non-dominated solutions, a bi-objective GA was employed.

In all of the studies reviewed so far, in order to simply the CS problem, it is assumed that the cycle is not allowed in the process routing of parts. While, in practice, parts can visit some machines more than once in their process routing. In the literature, parts having such a property are known as reentrant parts. By considering this property, Elmi et al. (2011) addressed a problem for the CS in a job shop CMS. It was assumed that parts belonging to a part family have identical setup times. A Simulated Annealing (SA) was employed to minimize the makespan. It should be noted that their problem did not include the RS problem. Li et al. (2016) assumed that all the machines are single-processing machines, except one that is batch-processing. It was also assumed that all the parts are EEs and parts belonging to a part family have identical setup times. Given the handling times between the cells and by considering reentrant parts, they presented an integrated mathematical model for the CS and RS problems. An Ant Colony Optimization (ACO) was applied to minimize the makespan.

In most existing studies in the context of CS, including those reviewed above, the researchers assume that the CF is pre-specified, and consider it as an input to the scheduling problem. Therefore, in some circumstances, the final schedule may not be efficient, due to its high dependence to the initial cell configuration. To overcome this difficulty, some researchers have investigated the CS and CF in an integrative manner, in order to get a better CMS design. Ghezavati and Saidi-Mehrabad(2010) addressed an integrated mathematical model for the CF and scheduling problems assuming that the processing times are stochastic. To deal with uncertainty, a scenario-based approach was applied to calculate the expected tardiness, subcontracting and resource underutilization costs. For simplicity, they made an unrealistic assumption by which EEs are eliminated using outsourcing. An algorithm combining the SA with the GA was employed to solve the problem.

Wang et al. (2010) presented an integrated mathematical model for the CF and CS by allowing machine duplication and considering traveling times between cells. It was assumed that parts are processed and transferred in batches. They solved the problem via an SS with the objective of minimizing the total tardiness penalty cost of batches. Rafiei et al. (2016) proposed a mathematical model to address the CF and CS problems concurrently. The proposed model minimizes the handling costs plus the operational costs per production cycle. For simplicity, they assumed that the operational cost per unit time is a known constant, and then, to estimate the operational costs, they multiplied it by the makespan. They solved the problem by a hybrid GA/SA-based algorithm. Liu et al. (2016) investigated the joint problem of CF and CS with worker assignment, multiple routings, and reentrant parts. A Discrete Bacteria Foraging Algorithm (DBFA) was used to solve the problem with the objective of minimizing the sum of handling, fixed and operating costs.

Besides the CS, some researchers have incorporated the Cell Layout (CL) in the CF problem. However, to prevent the problem from being too complex, the authors ignored some important issues like multiple routings, setup times/costs, and operational costs. In this respect, Wu et al. (2007) proposed an integrated CF, scheduling and layout problem in which the objective was to minimize the total flow time. It was assumed that the handling time of parts depends on the distance between the machines. They employed a GA to solve the problem. By allowing cycle in the process routing of parts, Arkat et al. (2012) presented a mathematical model to simultaneously identify the CF, its layout, and scheduling. A GA aiming at minimizing the total handling cost, as well as the makespan, was employed to solve the problem. Also, in another study, Arkat et al. (2012) modified the model presented by Arkat et al. (2012) to minimize either handling costs or the total flow time of parts. A GA was employed to solve the problem.

According to the above survey, some shortcomings can be identified in the attempted problems. In what follows, these shortcomings are discussed, and with respect to them, the contributions of this research are given.

- As it was mentioned earlier, in most current studies, the CS is investigated after the CF process. In order to achieve a more efficient design, in this paper, these two problems are integrated.
- Consideration of multiple process routings in the CS problem may lead to plans that have shorter completion times. Nevertheless, only a few studies have investigated the CS problem with multiple process routings. Therefore, this issue is addressed in the proposed problem.
- Once multiple process routings are taken into consideration, the processing and setup costs of each operation may differ on machines. Furthermore, in practice, the cycle may exist in the process routing of parts (reentrant parts). Despite the importance of these issues, most researchers have ignored them in the CS problem. So, the proposed problem is formulated by considering these issues.

_____

- Almost all researchers have studied CS problems under the assumption that each operation can be processed on at most one machine at a time. When multiple process routings are taken into consideration, and the processing requirement is considered as a total demand in production planning, the amount of workload required to be carried out for each operation can be distributed on an alternative set of machines. In the literature, such an approach is referred to as job splitting. In this study, to achieve shorter completion times, job splitting is also considered.

- Although in practice conflicting objectives are involved in the CS and CF problems, usually one objective is considered by the attempted problems. researchers. To cover this aspect, different criteria are reflected in the problem.

According to the explanations given above, in this papers, an integrated CF and CS problem is addressed while considering multiple process routings, reentrant parts, and job splitting. This problem is about finding the assignment of machines to the cells (i.e., CF), and assignment of operations to the machines (i.e., RS), determining the amount of workload that is dedicated to each machine (i.e., job splitting), and obtaining the sequence of operations on machines (i.e., CS). To the best of the authors' knowledge, this is the first research that incorporates all these decisions into one problem. The problem is formulated as a Mixed-Integer Program (MIP) so as to minimize the sum of processing, setup and inter-cell handling costs, plus the total weighted flow time of parts. Due to the computational complexity of the problem, two heuristics are suggested to find good solutions in a short computational time. Finally, by conducting computational experiments the performance of the heuristics is evaluated.

## 2. Problem statement and formulation

The problem addressed in this research takes several important sub-problems from the design process of the CMS into account. Decision making regarding the assignment of a set of machines to a set of cells by considering the maximum number of machines allowed in a cell is one of them. The other one is the scheduling of parts on machines in the presence of multiple process routings. The set of parts, as well as the set of operations required to complete each part, are known in advance. The processing requirement of parts is considered as a total demand in the planning horizon. For the sake of higher resource utilization and also shorten the completion time of parts, the workload required to be carried out for each operation is allowed to be distributed on the permissible set of machines. This results in two other sub-problems that one of them deals with the allocation of machines to each operation, i.e., RS problem, and the other one deals with how to distribute the workload of each operation between the selected set of machines. The scheduling approach is part-level, meaning that once part $i$ is scheduled before part $i'$, on the identical machines, all the operations of part $i$ must completed before those of part $i'$. According to these explanation, two conflicting objectives

are taken into consideration. One objective is to minimize the sum of processing, setup and, inter-cell transfer costs, and the other is to minimize the total weighted flow time of parts. A normalized weighted sum method is used to combine the objective functions.

## 2.1. Notations

**Indices and sets:**

$i, i' \in P$ — set of parts ($P = \{1, 2, \ldots, p\}$, where $p$ is the number of parts)

$j, j' \in O_i$ — set of operations of part $i$ ($O_i = \{1, 2, \ldots, o_i\}$, where $o_i$ is the number of operations of part $i$)

$k, k' \in M$ — set of machines ($M = \{1, 2, \ldots, m\}$, where $m$ is the number of machines)

$l \in C$ — set of cells ($C = \{1, 2, \ldots, c\}$, where $c$ is the number of cells)

**Parameters and decision variables:**

$d_i$ — demand of part $i$

$w_i$ — importance weight paid to the completion time of part $i$

$a_{ijk}$ — =1 if operation $j$ of part $i$ can be done on machine $k$; 0 otherwise

$t_{ijk}^P$ — processing time of operation $j$ of part $i$ on machine $k$

$c_{ijk}^P$ — unit processing cost of operation $j$ of part $i$ on machine $k$

$c_{ijk}^S$ — setup cost for processing operation $j$ of part $i$ on machine $k$

$c_i^E$ — unit inter-cell transfer cost of part $i$

$NM$ — maximum number of machines allowed to be assigned to a cell

$BM$ — a large enough number

$\alpha$ — weighting factor for unifying objectives

$CT$ — total weighted flow time

$PC$ — total processing cost

$SC$ — total setup cost

$HC$ — total inter-cell transfer cost

$f_1$ — estimated lower bound on the sum of processing, setup and handling costs

$f_2$ — estimated lower bound on the weighted completion time

$TC$ — weighted objective function

$x_{kl}$ — =1 if machine $k$ is assigned to cell $l$; 0 otherwise

$y_{ijk}$ — =1 if operation $j$ of part $i$ is carried out on machine $k$

$z_{ii'}$ — =1 if part $i'$ is scheduled after part $i$; 0 vice versa

$q_{ijk}$ — amount of workload dedicated to machine $k$ for processing operation $j$ of part $i$

$t_{ijk}$ — completion time of operation $j$ of part $i$ on machine $k$

$T_i$ — flow time of part $i$, i.e., $T_i = \max_{k \in M} \{t_{ijk} \,|\, j = o_i \text{ and } a_{ijk} = 1\}$

## 2.2. Mathematical model

On the basis of the explanations given above, the suggested problem is formulated as the following MIP.

Cellular Manufacturing Scheduling in the Presence of Multiple Process Routings and Considering Job Splitting

_____

$$\min TC = \alpha \frac{(PC + SC + HC)}{f_1} + (1 - \alpha)\frac{CT}{f_2}. \tag{1}$$

Subject to:

$$HC = \sum_{i \in P}\sum_{\substack{j \in O_i \\ j \neq 1}}\sum_{l \in C} c_i^E \max\left\{0, \sum_{k \in M}\left(a_{ijk}p_{ijk} - a_{i(j-1)k}p_{i(j-1)k}\right)x_{kl}\right\}, \tag{2}$$

$$SC = \sum_{i \in P}\sum_{j \in O_i}\sum_{k \in M} c_{ijk}^S y_{ijk}, \tag{3}$$

$$PC = \sum_{i \in P}\sum_{j \in O_i}\sum_{k \in M} c_{ijk}^P q_{ijk}, \tag{4}$$

$$CT = \sum_{i \in P} w_i T_i, \tag{5}$$

$$\sum_{k \in M} a_{ijk}q_{ijk} = d_i, \qquad i \in P, \qquad j \in O_i, \tag{6}$$

$$d_i y_{ijk} - q_{ijk} \geq 0, \qquad i \in P, \qquad j \in O_i, \qquad k \in M, \qquad a_{ijk} = 1, \tag{7}$$

$$\sum_{l \in C} x_{kl} = 1, \qquad k \in M, \tag{8}$$

$$\sum_{k \in M} x_{kl} \leq NM, \qquad l \in C, \tag{9}$$

$$t_{ijk} \geq t_{i'j'k} + t_{ijk}^P q_{ijk} - BM(2 - z_{i'i} - y_{ijk}), \qquad i, i' \in P, \qquad i > i', \qquad j \in O_i,$$
$$j' \in O_{i'}, \qquad k \in M, \qquad a_{ijk}a_{i'j'k} = 1, \tag{10}$$

$$t_{ijk} \geq t_{i'j'k} + t_{ijk}^P q_{ijk} - BM(1 + z_{ii'} - y_{ijk}), \qquad i, i' \in P, \qquad i < i', \qquad j \in O_i,$$
$$j' \in O_{i'}, \qquad k \in M, \qquad a_{ijk}a_{i'j'k} = 1, \tag{11}$$

$$t_{ijk} \geq t_{i(j-1)k'} + t_{ijk}^P q_{ijk} - BM(1 - y_{ijk}), \qquad i \in P, \qquad j \in O_i, \qquad j \neq 1,$$
$$k, k' \in M, \qquad a_{ijk}a_{i(j-1)k'} = 1, \tag{12}$$

$$t_{ijk} \geq t_{ijk}^P q_{ijk}, \qquad i \in P, \qquad j = 1, \qquad k \in M, \qquad a_{ijk} = 1, \tag{13}$$

$$T_i \geq t_{ijk}, \qquad i \in P, \qquad j = o_i, \qquad k \in M, \qquad a_{ijk} = 1, \tag{14}$$

$$T_i \geq 0, \qquad i \in P, \tag{15}$$

$$t_{ijk} \geq 0, \qquad i \in P, \qquad j \in O_i, \qquad k \in M, \tag{16}$$

$$q_{ijk} \geq 0, \qquad i \in P, \qquad j \in O_i, \qquad k \in M, \tag{17}$$

$$y_{ijk} \in \{0,1\}, \qquad i \in P, \qquad j \in O_i, \qquad k \in M, \tag{18}$$

$$x_{kl} \in \{0,1\}, \qquad k \in M, \qquad l \in C, \tag{19}$$

$$z_{ii'} \in \{0,1\}, \qquad i, i' \in P, \qquad i' > i. \tag{20}$$

In the model above, objective function (10) minimizes the sum of processing, setup, and inter-cell handling costs plus the total weighted flow time of parts. It should be noted that in this objective function, the importance paid to each objective can be adjusted by giving an appropriate value to $\alpha$. Eqs. (2)–(5) respectively correspond to the calculation of the inter-cell handling cost, setup cost, processing cost and total weighted flow time. Constraint set (6) states that for each

Kamran Forghani, Seyyed Mohammad Taghi Fatemi-Ghomi

operation of a part, the sum of workload allocated to the permissible set of machines must be equal to the demand of the corresponding part. Constraint set (7) represents that if decision variable $q_{ijk}$ takes a positive value, binary variable $y_{ijk}$ must be equal to 1. Constraint set (8) ensures that each machine is assigned to a cell, and constraint set (9) is to avoid the assignment of more than $NM$ machines to each cell. Constraint sets (10) and (11), ensure that the completion time of operations on the same machines are determined according to the scheduled plan. Constraint sets (12) and (13) ensure that for each part, the sequence of operations is taken into consideration in the completion times. Constraint set (14) ensures that the flow time of each part is greater than the completion time of the last operation of the corresponding part on the permissible set of machines; in fact, this constraint is equivalent with $T_i \geq \max_{k \in M}\{t_{ijk}|j = o_i \text{ and } a_{ijk} = 1\}$. Finally, constraint sets (15)–(20) impose the non-negativity and binary requirements on the decision variables.

**2.3. Correctness of the model**

In this section, the correctness of Eq. (2), as well as constraint sets (10)–(13), are investigated (verifying the correctness of the other constraints in the model above is trivial).

**Lemma 1.** Eq. (2) correctly returns the total inter-cell transfer cost.

**Proof.** Let $S_l$ denote the set of machines assigned to cell $l$. Thus, $S_l = \{k|k \in M, x_{kl} = 1\}$ for $l \in C$. Let also $Q_{ijl}$ is defined by $Q_{ijl} = \sum_{k \in S_l} q_{ijk}$ for $i \in P$, $j \in O_i$, $l \in C$. Now, for two consecutive operations $j - 1$ and $j$ of part $i$ ($j \in O_i$, $j \neq 1$), the number of transfers (including the inter- and intra-cell transfers) required to move part $i$ from cell $l$ to the other cells and cell $l$ itself, equals $E_{ijl} + F_{ijl} = Q_{i(j-1)l}$, where $E_{ijl}$ and $F_{ijl}$ respectively stand for the corresponding number of inter- and intra-cell transfers, see Figure 1. Clearly, for the sake of feasibility, inequalities $F_{ijl} \leq Q_{i(j-1)l}$ and $F_{ijl} \leq Q_{ijl}$ should hold for $i \in P$, $j \in O_i$, $j \neq 1$, $l \in C$. By these definitions, the total inter-cell transfer cost can be determined by $HC = \sum_{i \in P}\sum_{\substack{j \in O_i \\ j \neq 1}}\sum_{l \in C} c_i^E(Q_{i(j-1)l} - F_{ijl})$. As the objective is to minimize $HC$, we can set $F_{ijl} = \min\{Q_{i(j-1)l}, Q_{ijl}\}$. Thus, we have $HC = \sum_{i \in P}\sum_{\substack{j \in O_i \\ j \neq 1}}\sum_{l \in C} c_i^E(Q_{i(j-1)l} - \min\{Q_{i(j-1)l}, Q_{ijl}\})$. Finally, it is easy to conclude that $HC = \sum_{i \in P}\sum_{\substack{j \in O_i \\ j \neq 1}}\sum_{l \in C} c_i^E \max\{0, Q_{ijl} - Q_{i(j-1)l}\}$, which is consistent with Eq. (2).
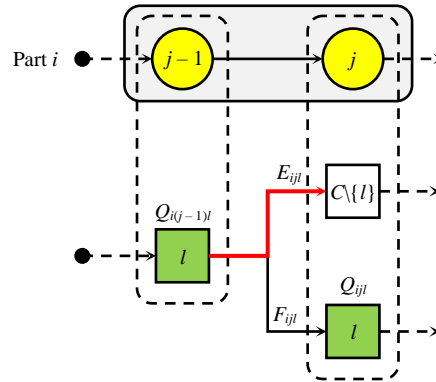
_____



**Figure 1.Material flow originating from cell $l$ based on two consecutive operations of a part**

**Lemma 2.**Constraint sets (10) and (11) correctly determine the completion time of operations of the same machines.

**Proof.** Let $S_{ik}$ denote the set of parts to be scheduled before part $i$ on machine $k$. So, according to the definition of decision variable $z_{ii'}$, we have $S_{ik} = \{i'|z_{ii'} = 0, i' \in P, i' > i\} \cup \{i'|z_{i'i} = 1, i' \in P, i' < i\}$ for $i, \in P, k \in M$. Now, if operation $j$ of part $i$ is not assigned to machine $k$, i.e., $y_{ijk} = 0$, the completion time of corresponding operation on machine $k$ must be zero, i.e., $t_{ijk} = 0$; otherwise it must be greater than the time required to process the corresponding operation on machine $k$, plus the maximum of the completion time of the operations scheduled before operation $j$ of part $i$ on machine k, i.e., $t_{ijk} \geq t_{ijk}^P q_{ijk} + \max_{i' \in S_{ik}, j' \in O_{i'}} \{t_{i'j'k}\}$. As in the proposed problem a positive coefficient of $t_{ijk}$'s is minimized, see constraint sets (10), (5) and (14), such requirements can be meet by imposing constraint $t_{ijk} \geq y_{ijk} \left( t_{ijk}^P q_{ijk} + \max_{i' \in S_{ik}, j' \in O_{i'}} \{t_{i'j'k}\} \right)$ for $i \in P$, $j \in O_i$, $k \in M$. Finally it is correct to derive $t_{ijk} \geq y_{ijk} \left( t_{ijk}^P q_{ijk} + t_{i'j'k} \right)$ for $i, \in P$, , $j \in O_i, k \in M, i' \in S_{ik}, j' \in O_{i'}$ which is consistent with constraint sets (10) and (11).

**Lemma 3.**Constraint sets (12) and (13) correctly takes the sequence of operations into account.

**Proof.** Considering two consecutive operations $j - 1$ and $j$ of part $i$ ($j \in O_i, j \neq 1$), if operation $j$ is not assigned to machine $k$, we must derive $t_{ijk} = 0$; otherwise, the completion time of operation $j$ of part $i$ on machine $k$ must be greater than its processing time on machine $k$, plus the maximum of the completion time of preceding operation on any machine, i.e., $t_{ijk} \geq t_{ijk}^P q_{ijk} + \max_{k' \in M} \{t_{i(j-1)k'}\}$. As in

**279**

the proposed problem a positive coefficient of $t_{ijk}$'s is minimized, see constraints (10), (5) and (14), these requirements can be satisfied by imposing constraint

$$t_{ijk} \geq y_{ijk}\left(t^P_{ijk}q_{ijk} + \max_{k'\in M}\{t_{i(j-1)k'}\}\right) \quad \text{for} \quad i \in P,\ j \in O_i, j \neq 1,\ k \in M.$$ To resolve max operator in this constraint set, we can write $t_{ijk} \geq y_{ijk}(t^P_{ijk}q_{ijk} + t_{i(j-1)k'})$ for $i \in P, j \in O_i, j \neq 1, k, k' \in M$, which is equivalent with constraint set (12). Finally, as the first operation of each part has not a preceding operation, we only need to impose constraint set (13).

## 2.4. Linearized mathematical model

In order to effectively solve the proposed problem using a MIP solver, non-linear term $\max\{0, \sum_{k\in M}(a_{ijk}p_{ijk} - a_{i(j-1)k}p_{i(j-1)k})x_{kl}\}$ in Eq. (2)should be linearized. In doing so, two sets of auxiliary variables $\gamma_{ijl}$ and $\beta_{ijkl}$ are introduced. It should be noted that $\gamma_{ijl}$'s are positive variables and are $\beta_{ijkl}$'s are free variables. Now, non-linear term $(a_{ijk}p_{ijk} - a_{i(j-1)k}p_{i(j-1)k})x_{kl}$ is replaced with auxiliary variable $\beta_{ijkl}$ and constraint sets (22) and (23) are included in the model. Finally, to complete the linearization of the model, non-linear term $\max\{0, \sum_{k\in M}\beta_{ijkl}\}$ is replaced with auxiliary variable $\gamma_{ijl}$, and also, constraint sets (24) and (25) are added to the model.

min (10).
Subject to: (3)–(20),

$$HC = \sum_{i\in P}\sum_{\substack{j\in O_i \\ j\neq 1}}\sum_{l\in C} c^E_i \gamma_{ijl} \tag{21}$$

$$\beta_{ijkl} \geq a_{ijk}p_{ijk} - a_{i(j-1)k}p_{i(j-1)k} - d_i(1 - x_{kl}), \quad i \in P, \quad j \in O_i, \quad j \neq 1, \quad k \in M, \quad l \in C, \tag{22}$$

$$\beta_{ijkl} \geq -d_i x_{kl}, \quad i \in P, \quad j \in O_i, \quad j \neq 1, \quad a_{ijk} + a_{i(j-1)k} \geq 1, \quad k \in M, \quad l \in C, \tag{23}$$

$$\gamma_{ijl} \geq \sum_{k\in M | a_{ijk}\lor a_{i(j-1)k}=1} \beta_{ijkl}, \quad i \in P, \quad j \in O_i, \quad j \neq 1, \quad l \in C, \tag{24}$$

$$\gamma_{ijl} \geq 0, \quad i \in P, \quad j \in O_i, \quad j \neq 1, \quad l \in C, \tag{25}$$

## 3. Heuristic methods

The original problem involves the following sub-problems: Cell Formation Problem (CFP), Routing Selection Problem (RSP), Job Splitting Problem (JSP) and Scheduling Problem (SP). According to these sub-problems, some combined sub-problems are defined. Then, two heuristics iterating between these combined sub-problems are presented. Hereafter, each combined sub-problem is denoted by using the name of corresponding sub-problems separated by sign '-'. For instance, SP-JSP means that the scheduling and job splitting problems are simultaneously taken into consideration in a combined sub-problem, and the decision variables in

connection to the other sub-problems, in this case, the CF and RS problems, are
assumed to be fixed.

At the beginning of each heuristic, and also, at the beginning of solving the
original problem, we need to know the value of $f_1$ and $f_2$. In this regard, the
following two problems needs to be solved.

**Initial-CF-RSP-JSP:**

Outputs: $f_1$ and $\bar{x}_{kl}, \bar{y}_{ijk}, \bar{q}_{ijk}$ for $i \in P, \ j \in O_i, \ k \in M, \ l \in C$.

$$f_1 = \min PC + SC + HC. \tag{26}$$

Subject to: (3), (4), (6)–(9), (17)–(19), (21)–(25).

**Initial-SP-RSP-JSP:**

Outputs: $f_2$ and $\bar{z}_{ii'}, \bar{y}_{ijk}, \bar{q}_{ijk}$ for $i, i' \in P, \ i' > i, \ j \in O_i, \ k \in M$.

$$f_2 = \min CT. \tag{27}$$

Subject to: (5)–(7), (10)–(18), (20).

### 3.1. Heuristic I

The first heuristic implements CFP-RSP-JSP and SP-JSP. The corresponding
mathematical models, as well as their inputs and outputs, are given below.

**CFP-RSP-JSP:**

Inputs: $\bar{z}_{ii'}$ for $i, i' \in P, \ i' > i$.

Outputs: $\bar{x}_{kl}, \bar{y}_{ijk}, \bar{q}_{ijk}$ for $i \in P, \ j \in O_i, \ k \in M, \ l \in C$.

min (10).

Subject to: (3)–(9), (12)–(19), (21)–(25),

$$t_{ijk} \geq t_{i'j'k} + t^P_{ijk} q_{ijk} - BM(1 - y_{ijk}), i, i' \in P, \quad i > i', \quad j \in O_i,$$
$$j' \in O_{i'}, \quad k \in M, \quad a_{ijk} a_{i'j'k} = 1, \quad \bar{z}_{i'i} = 1, \tag{28}$$

$$t_{ijk} \geq t_{i'j'k} + t^P_{ijk} q_{ijk} - BM(1 - y_{ijk}), i, i' \in P, \quad i < i', \quad j \in O_i,$$
$$j' \in O_{i'}, \quad k \in M, \quad a_{ijk} a_{i'j'k} = 1, \quad \bar{z}_{ii'} = 0. \tag{29}$$

**SP-JSP:**

Inputs: $\bar{x}_{kl}, \bar{y}_{ijk}$ for $i \in P, \ j \in O_i, \ k \in M, \ l \in C$.

Outputs: $\bar{z}_{ii'}, \bar{q}_{ijk}$ for $i, i' \in P, \ i' > i, \ j \in O_i, \ k \in M$.

$$\min TC = \alpha \frac{(\overline{SC} + PC + HC)}{f_1} + (1 - \alpha) \frac{CT}{f_2}. \tag{30}$$

Subject to: (4)–(6), (13)–(17), (20), (21), (25),

$$d_i \bar{y}_{ijk} - q_{ijk} \geq 0, i \in P, \quad j \in O_i, \quad k \in M, \quad a_{ijk} = 1, \tag{31}$$

$$t_{ijk} \geq t_{i'j'k} + t^P_{ijk} q_{ijk} - BM(1 - z_{i'i}), i, i' \in P, \quad i > i', \quad j \in O_i,$$
$$j' \in O_{i'}, \quad k \in M, \quad a_{ijk} a_{i'j'k} = 1, \quad \bar{y}_{ijk} = 1, \tag{32}$$

$$t_{ijk} \geq t_{i'j'k} + t^P_{ijk} q_{ijk} - BM.z_{ii'}, i, i' \in P, \quad i < i', \quad j \in O_i,$$
$$j' \in O_{i'}, \quad k \in M, \quad a_{ijk} a_{i'j'k} = 1, \quad \bar{y}_{ijk} = 1, \tag{33}$$

$$t_{ijk} \geq t_{i(j-1)k'} + t^P_{ijk} q_{ijk}, i \in P, \quad j \in O_i, \quad j \neq 1, \quad k, k' \in M,$$
$$a_{ijk} a_{i(j-1)k'} = 1, \quad \bar{y}_{ijk} = 1, \tag{34}$$

_____

$$\gamma_{ijl} \geq \sum_{k \in M} \bar{z}_{kl}\big(a_{ijk}q_{ijk} - a_{i(j-1)k}q_{i(j-1)k}\big), \qquad i \in P, \qquad j \in O_i,$$
$$j \neq 1, \qquad l \in C, \tag{35}$$

where

$$\overline{SC} = \sum_{i \in P} \sum_{j \in O_i} \sum_{k \in M} c_{ijk}^S \bar{y}_{ijk}. \tag{36}$$

Now, the steps of Heuristic I are summarized as follows:

**Step 1.** Set $N = 0$, $N' = 0$ and $\overline{TC} = \infty$. Solve (Initial-CFP-RSP-JSP) and obtain $f_1$ and $\bar{x}_{kl}, \bar{y}_{ijk}, \bar{q}_{ijk}$ for $i \in P$, $j \in O_i$, $k \in M$, $l \in C$. Solve (Initial-SP-JSP) and obtain $f_2$ and $\bar{z}_{ii'}, \bar{q}_{ijk}$ for $i, i' \in P$, $i' > i$, $j \in O_i$, $k \in M$. If $w \leq 0.5$, go to Step 2; otherwise go to Step 3.

**Step 2.** Set $N = N + 1$ and $N' = N' + 1$. Solve (CF-RSP-JSP) and obtain $TC^*$ and $\bar{x}_{kl}, \bar{y}_{ijk}, \bar{q}_{ijk}$ for $i \in P$, $j \in O_i$, $k \in M$, $l \in C$. If $TC^* < \overline{TC}$, set $\overline{TC} = TC^*$, $N' = 0$ and go to Step 3. If $N' = 2$, stop; otherwise go to Step 3.

**Step 3.** Set $N = N + 1$ and $N' = N' + 1$. Solve (SP-JSP) and obtain $TC^*$ and $\bar{z}_{ii'}, \bar{q}_{ijk}$ for $i, i' \in P$, $i' > i$, $j \in O_i$, $k \in M$. If $TC^* < \overline{TC}$, set $\overline{TC} = TC^*$, $N' = 0$ and go to Step 2. If $N' = 2$, stop; otherwise go to Step 2.

### 3.2. Heuristic II

The second heuristic iterates between SP-RSP-JSP and CF-RSP-JSP. The mathematical model corresponding to CF-RSP-JSP is given in the previous subsection, and the mathematical model associated with SP-RSP-JSP, as well as its inputs and outputs, are given below.

**SP-RSP-JSP:**
Inputs: $\bar{x}_{kl}$ for $k \in M$, $l \in C$.
Outputs: $\bar{z}_{ii'}, \bar{y}_{ijk}, \bar{q}_{ijk}$ for $i, i' \in P$, $i' > i$, $j \in O_i$, $k \in M$.
min (10).
Subject to: (3)–(7), (10)–(18), (20), (21), (25), (35).

The steps of Heuristic II are also summarized as follows:

**Step 1.** Set $N = 0$, $N' = 0$ and $\overline{TC} = \infty$. Solve (Initial-CFP-RSP-JSP) and obtain $f_1$ and $\bar{x}_{kl}, \bar{y}_{ijk}, \bar{q}_{ijk}$ for $i \in P$, $j \in O_i$, $k \in M$, $l \in C$. Solve (Initial-SP-JSP) and obtain $f_2$ and $\bar{z}_{ii'}, \bar{q}_{ijk}$ for $i, i' \in P$, $i' > i$, $j \in O_i$, $k \in M$. If $w \leq 0.5$, go to Step 2; otherwise go to Step 3.

_____

**Step 2.** Set $N = N + 1$ and $N' = N' + 1$. Solve (CF-RSP-JSP) and obtain $TC^*$ and $\bar{x}_{kl}, \bar{y}_{ijk}, \bar{q}_{ijk}$ for $i \in P,\ j \in O_i,\ k \in M,\ l \in C$. If $TC^* < \overline{TC}$, set $\overline{TC} = TC^*$, $N' = 0$ and go to Step 3. If $N' = 2$, stop; otherwise go to Step 3.

**Step 3.** Set $N = N + 1$ and $N' = N' + 1$. Solve (SP-RSP-JSP) and obtain $TC^*$ and $\bar{z}_{ii'}, \bar{y}_{ijk}, \bar{q}_{ijk}$ for $i, i' \in P,\ i' > i,\ j \in O_i,\ k \in M$. If $TC^* < \overline{TC}$, set $\overline{TC} = TC^*$, $N' = 0$ and go to Step 2. If $N' = 2$, stop; otherwise go to Step 2.

## 4. Computational results

To verify the performance of the proposed heuristics, eleven instances in different sizes are solved. The proposed heuristics, as well as the original model, were coded in Gams programming language, and Cplex was chosen as the MIP solver. Computations in this study are carried out on a PC having Intel Core i7 4790k CPU (with 4 cores, 8 threads and running at 4 GHz) and 16 GB of RAM. The dataset of these instances is generated at random. Table 1 gives the specification of the instances, as well as the necessary information that is required to generate the parameters randomly.

**Table 1. Characteristics of the test problems**

| Instance # | $p \times m$ | $c$ | $NM$ | $P_0$ | $o_i$ | $f_1$ | $f_2$ |
|---|---|---|---|---|---|---|---|
| 1 | $20 \times 15$ | 4 | 4 | 0.6 | $UI(1,5)$ | 4492 | 1416.247 |
| 2 | $25 \times 15$ | 4 | 4 | 0.6 | $UI(1,5)$ | 4870 | 2002.274 |
| 3 | $25 \times 20$ | 4 | 6 | 0.5 | $UI(1,5)$ | 4499 | 1666.066 |
| 4 | $30 \times 15$ | 4 | 4 | 0.5 | $UI(1,5)$ | 5235 | 2454.451 |
| 5 | $30 \times 20$ | 4 | 6 | 0.5 | $UI(1,5)$ | 4675 | 2159.272 |
| 6 | $40 \times 15$ | 4 | 4 | 0.5 | $UI(1,5)$ | 7531 | 4678.828 |
| 7 | $40 \times 20$ | 4 | 6 | 0.5 | $UI(1,6)$ | 9081 | 5020.156 |
| 8 | $40 \times 25$ | 5 | 6 | 0.5 | $UI(2,6)$ | 11203 | 5639.526 |
| 9 | $50 \times 15$ | 4 | 4 | 0.5 | $UI(2,5)$ | 13574 | 8103.928 |
| 10 | $50 \times 20$ | 4 | 6 | 0.5 | $UI(2,5)$ | 12472 | 6728.101 |
| 11 | $50 \times 25$ | 5 | 6 | 0.5 | $UI(2,6)$ | 14921 | 6941.271 |

Once parameter $o_i$ was randomly generated according to Table 1, parameter $a_{ijk}$ is generated by $a_{ijk} = \begin{cases} 1 & (U(0,1) < 1 - P_0{}^m)\text{or}(k = k_0) \\ 0 & \text{otherwise} \end{cases}$, $i \in P,\ j \in O_i,\ k \in M$, where $P_0$ is a constant that is selected according to Table 1, and $k_0$ is an integer number that is randomly chosen using discreet uniform distribution as $k_0 = UI(1, m)$. The remaining parameters are also randomly generated using the following information: $d_i = 10 \times UI(5,15)$, $c_i^E = 0.1 \times UI(5,15)$, $w_i = 0.1 \times UI(2,10)$, $t_{ijk}^P = 0.1 \times UI(2,10)$, $c_{ijk}^P = 0.1 \times UI(2,10)$ and $c_{ijk}^S = UI(10,20)$ for $i \in P,\ j \in O_i,\ k \in M$ and $a_{ijk} = 1$. It should be noted that the values reported for $f_1$ and $f_2$, in Table 1, have been obtained by solving the models give in section 'Heuristic methods', i.e., Initial-CF-RSP-JSP and Initial-SP-RSP-JSP. In order to

find a set of non-dominated solutions, each instance is solved for different values of $\alpha$, ranging from 0.1 to 0.9.The time limit for solving each sub-model within the heuristic methods is set to 100 seconds, also for the original model, it is set to 1000 seconds. The computational results are summarized in Tables 2–5. In these tables, 'Ite.' stands for the number of sub-models solved within the corresponding heuristic, and 'Gap' denotes the relative optimality gap obtained by Cplex.

**Table 2.Summary of computation results for instances #1–3**

| $\alpha$ | Instance #1 | | | | | | Instance #2 | | | | | | Instance #3 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Heuristic I | | Heuristic II | | Cplex | | Heuristic I | | Heuristic II | | Cplex | | Heuristic I | | Heuristic II | | Cplex | |
| | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap |
| 0.1 | 1.019 | 2 | 1.019 | 2 | 1.051 | 0.260 | 1.026 | 2 | 1.025 | 2 | 1.028 | 0.236 | 1.030 | 2 | 1.030 | 2 | 1.042 | 0.239 |
| 0.2 | 1.037 | 2 | 1.037 | 2 | 1.056 | 0.241 | 1.043 | 2 | 1.043 | 2 | 1.062 | 0.222 | 1.057 | 2 | 1.057 | 2 | 1.063 | 0.221 |
| 0.3 | 1.054 | 2 | 1.054 | 2 | 1.074 | 0.227 | 1.045 | 6 | 1.056 | 2 | 1.055 | 0.204 | 1.081 | 2 | 1.078 | 3 | 1.075 | 0.193 |
| 0.4 | 1.064 | 2 | 1.050 | 4 | 1.093 | 0.232 | 1.053 | 4 | 1.062 | 3 | 1.079 | 0.222 | 1.098 | 2 | 1.095 | 4 | 1.104 | 0.193 |
| 0.5 | 1.069 | 2 | 1.053 | 3 | 1.093 | 0.205 | 1.060 | 4 | 1.054 | 3 | 1.056 | 0.156 | 1.108 | 4 | 1.106 | 3 | 1.109 | 0.196 |
| 0.6 | 1.065 | 7 | 1.052 | 4 | 1.094 | 0.173 | 1.086 | 5 | 1.054 | 3 | 1.054 | 0.128 | 1.119 | 5 | 1.100 | 2 | 1.104 | 0.169 |
| 0.7 | 1.054 | 5 | 1.053 | 2 | 1.060 | 1.060 | 1.075 | 4 | 1.053 | 2 | 1.051 | 0.114 | 1.109 | 3 | 1.090 | 2 | 1.098 | 0.154 |
| 0.8 | 1.038 | 4 | 1.038 | 2 | 1.056 | 0.087 | 1.053 | 4 | 1.042 | 3 | 1.042 | 0.055 | 1.073 | 2 | 1.068 | 2 | 1.076 | 0.116 |
| 0.9 | 1.021 | 4 | 1.021 | 2 | 1.024 | 0.017 | 1.029 | 4 | 1.027 | 2 | 1.027 | 0.012 | 1.036 | 2 | 1.036 | 2 | 1.044 | 0.057 |

**Table 3.Summary of computation results for instances #4 and #5**

| $\alpha$ | Instance #4 | | | | | | Instance #5 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Heuristic I | | Heuristic II | | Cplex | | Heuristic I | | Heuristic II | | Cplex | |
| | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap |
| 0.1 | 1.028 | 2 | 1.026 | 4 | 1.043 | 0.244 | 1.029 | 2 | 1.029 | 4 | 1.034 | 0.321 |
| 0.2 | 1.052 | 4 | 1.054 | 3 | 1.053 | 0.230 | 1.062 | 2 | 1.062 | 2 | 1.063 | 0.290 |
| 0.3 | 1.072 | 4 | 1.076 | 3 | 1.098 | 0.225 | 1.085 | 2 | 1.085 | 3 | 1.087 | 0.281 |
| 0.4 | 1.092 | 3 | 1.092 | 3 | 1.094 | 0.218 | 1.096 | 4 | 1.096 | 3 | 1.097 | 0.244 |
| 0.5 | 1.105 | 2 | 1.102 | 3 | 1.118 | 0.194 | 1.111 | 2 | 1.108 | 3 | 1.113 | 0.219 |
| 0.6 | 1.153 | 4 | 1.111 | 4 | 1.105 | 0.166 | 1.134 | 5 | 1.106 | 2 | 1.092 | 0.181 |
| 0.7 | 1.117 | 4 | 1.103 | 2 | 1.085 | 0.120 | 1.108 | 6 | 1.093 | 3 | 1.101 | 0.163 |
| 0.8 | 1.080 | 2 | 1.076 | 2 | 1.072 | 0.094 | 1.080 | 4 | 1.074 | 2 | 1.106 | 0.136 |
| 0.9 | 1.040 | 2 | 1.040 | 2 | 1.040 | 0.012 | 1.046 | 3 | 1.043 | 2 | 1.043 | 0.033 |

**Table 4.Summary of computation results for instances #6–8**

| $\alpha$ | Instance #6 | | | | | | Instance #7 | | | | | | Instance #8 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Heuristic I | | Heuristic II | | Cplex | | Heuristic I | | Heuristic II | | Cplex | | Heuristic I | | Heuristic II | | Cplex | |
| | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap |
| 0.1 | 1.028 | 4 | 1.021 | 5 | 1.044 | 0.423 | 1.021 | 5 | 1.030 | 4 | 1.058 | 0.416 | 1.047 | 6 | 1.013 | 7 | 1.051 | 0.383 |
| 0.2 | 1.045 | 5 | 1.043 | 3 | 1.087 | 0.424 | 1.034 | 5 | 1.044 | 5 | 1.117 | 0.409 | 1.051 | 8 | 1.024 | 9 | 1.059 | 0.359 |
| 0.3 | 1.064 | 4 | 1.058 | 3 | 1.076 | 0.358 | 1.049 | 5 | 1.044 | 3 | 1.087 | 0.357 | 1.077 | 3 | 1.065 | 3 | 1.073 | 0.337 |
| 0.4 | 1.072 | 5 | 1.069 | 3 | 1.084 | 0.329 | 1.056 | 5 | 1.062 | 3 | 1.123 | 0.353 | 1.091 | 4 | 1.060 | 3 | 1.092 | 0.326 |
| 0.5 | 1.079 | 8 | 1.082 | 3 | 1.071 | 0.284 | 1.061 | 4 | 1.068 | 3 | 1.110 | 0.319 | 1.062 | 6 | 1.077 | 3 | 1.060 | 0.274 |
| 0.6 | 1.116 | 6 | 1.065 | 8 | 1.098 | 0.275 | 1.111 | 2 | 1.067 | 2 | 1.093 | 0.270 | 1.158 | 2 | 1.085 | 2 | 1.105 | 0.297 |
| 0.7 | 1.101 | 4 | 1.080 | 2 | 1.073 | 0.225 | 1.082 | 2 | 1.058 | 2 | 1.138 | 0.282 | 1.118 | 2 | 1.087 | 2 | 1.080 | 0.251 |
| 0.8 | 1.073 | 3 | 1.061 | 2 | 1.068 | 0.189 | 1.057 | 2 | 1.053 | 2 | 1.060 | 0.201 | 1.079 | 2 | 1.068 | 2 | 1.089 | 0.221 |
| 0.9 | 1.035 | 2 | 1.035 | 2 | 1.040 | 0.123 | 1.024 | 4 | 1.023 | 2 | 1.066 | 0.180 | 1.035 | 4 | 1.036 | 2 | 1.067 | 0.171 |

**Table 5.Summary of computation results for instances #9–11**

| $\alpha$ | Instance #9 | | | | | | Instance #10 | | | | | | Instance #11 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Heuristic I | | Heuristic II | | Cplex | | Heuristic I | | Heuristic II | | Cplex | | Heuristic I | | Heuristic II | | Cplex | |
| | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap | $TC$ | Ite. | $TC$ | Ite. | $TC$ | Gap |
| 0.1 | 1.015 | 8 | 1.041 | 11 | 1.030 | 0.523 | 1.035 | 6 | 1.043 | 5 | 1.095 | 0.505 | 1.081 | 4 | 1.055 | 5 | 1.022 | 0.460 |
| 0.2 | 1.073 | 8 | 1.056 | 9 | 1.045 | 0.496 | 1.063 | 4 | 1.059 | 3 | 1.133 | 0.467 | 1.068 | 10 | 1.058 | 7 | 1.162 | 0.487 |

| 0.3 | 1.040 | 12 | 1.058 | 13 | 1.078 | 0.461 | 1.036 | 7 | 1.061 | 5 | 1.066 | 0.437 | 1.050 | 15 | 1.062 | 9 | 1.112 | 0.430 |
| 0.4 | 1.057 | 11 | 1.097 | 9 | 1.074 | 0.418 | 1.073 | 6 | 1.065 | 9 | 1.086 | 0.411 | 1.063 | 10 | 1.042 | 10 | 1.056 | 0.364 |
| 0.5 | 1.051 | 5 | 1.065 | 5 | 1.088 | 0.393 | 1.082 | 6 | 1.091 | 3 | 1.077 | 0.364 | 1.072 | 10 | 1.074 | 5 | 1.094 | 0.358 |
| 0.6 | 1.060 | 4 | 1.070 | 4 | 1.052 | 0.343 | 1.095 | 4 | 1.068 | 4 | 1.099 | 0.350 | 1.095 | 7 | 1.072 | 4 | 1.110 | 0.339 |
| 0.7 | 1.045 | 4 | 1.069 | 2 | 1.071 | 0.316 | 1.067 | 4 | 1.061 | 2 | 1.102 | 0.320 | 1.095 | 5 | 1.076 | 2 | 1.139 | 0.330 |
| 0.8 | 1.041 | 2 | 1.036 | 2 | 1.131 | 0.318 | 1.044 | 4 | 1.047 | 2 | 1.074 | 0.268 | 1.059 | 6 | 1.055 | 4 | 1.079 | 0.258 |
| 0.9 | 1.021 | 2 | 1.020 | 2 | 1.068 | 0.249 | 1.023 | 2 | 1.027 | 2 | 1.083 | 0.245 | 1.029 | 4 | 1.028 | 2 | 1.043 | 0.179 |

From Tables 2–5 it is observed that Cplex could not solve any of the cases to optimality in less than 1000 seconds. Also, it is seen that by increasing the instance size, the relative optimality gap increases. In order evaluate the performance of the heuristics against each other, and also against the Cplex, the paired $t$-Test is used. In this way, the weighted objective value, $TC$, is considered a reference to compare the solutions. The comparison results are reported in Table 6. Also, for each instance, the sum of CPU times is illustrated in Figure2. As it is seen in Table 6, in all the cases the null hypothesis, $H_0$, is rejected with a very high confidence level. Therefore, it can be concluded that both the heuristics can produce better solution, in considerably less computational time, compared to Cplex. Also, it is seen that Heuristic II outperforms Heuristic I. However, according to Figure2, no significant difference is seen between the CPU time of the heuristics.
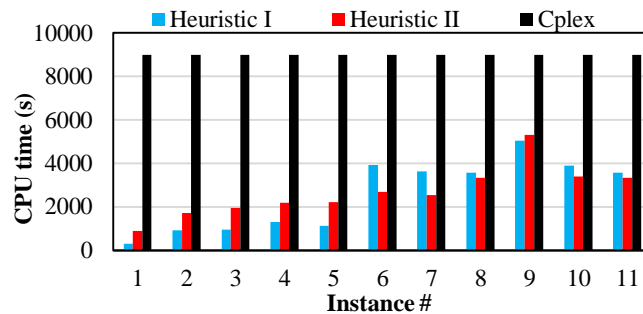


**Figure 2. Comparison between the CPU time of the solution methods**

**Table 6. Results of paired *t*-test for means**

| pairs | Cplex | Heuristic I | Cplex | Heuristic II | Cplex | Best* | Heuristic I | Heuristic II |
|---|---|---|---|---|---|---|---|---|
| mean | 1.0773 | 1.0651 | 1.0773 | 1.0592 | 1.0773 | 1.0565 | 1.0651 | 1.0592 |
| variance | 0.0008 | 0.0009 | 0.0008 | 0.0006 | 0.0008 | 0.0006 | 0.0009 | 0.0006 |
| observations | 99 | 99 | 99 | 99 | 99 | 99 | 99 | 99 |
| $df$ | 98 | | 98 | | 98 | | 98 | |
| $H_0$ | $\mu_{Cplex} < \mu_{Heuristic\,I}$ | | $\mu_{Cplex} < \mu_{Heuristic\,II}$ | | $\mu_{Cplex} < \mu_{Best}$ | | $\mu_{Heuristic\,I} < \mu_{Heuristic\,II}$ | |
| $H_1$ | $\mu_{Cplex} \geq \mu_{Heuristic\,I}$ | | $\mu_{Cplex} \geq \mu_{Heuristic\,II}$ | | $\mu_{Cplex} \geq \mu_{Best}$ | | $\mu_{Heuristic\,I} \geq \mu_{Heuristic\,II}$ | |
| test statistic | 4.2334 | | 7.2132 | | 8.3549 | | 3.6245 | |
| $p$-value | 2.6E-05 | | 5.8E-11 | | 2.21E-13 | | 0.0002 | |

* Best solution derived from Heuristic I and Heuristic II.

For each instance, the set of non-dominated solutions obtained by the proposed heuristics and Cplex are plotted in Figure3. From this figure, it is seen that in most

cases, the solution produced by Cplex is dominated by that obtained by the heuristics (see, for example, Instance #7). Regarding the number of non-dominated solutions contributing in Pareto frontier, Heuristic II has a better condition, then followed by Heuristic I and Cplex (see, for example, Figure 3, Instance #6). This demonstrates that Heuristic II is able to produce a wider variety of non-dominated solutions. Although, in terms of $TC$, Heuristic II outperforms Heuristic I, however, in Figure3, it is seen that in some cases the non-dominated solutions produced by Heuristic II are dominated by that of Heuristic I. Therefore, in order to attain better results, it would be better to implement both heuristics.
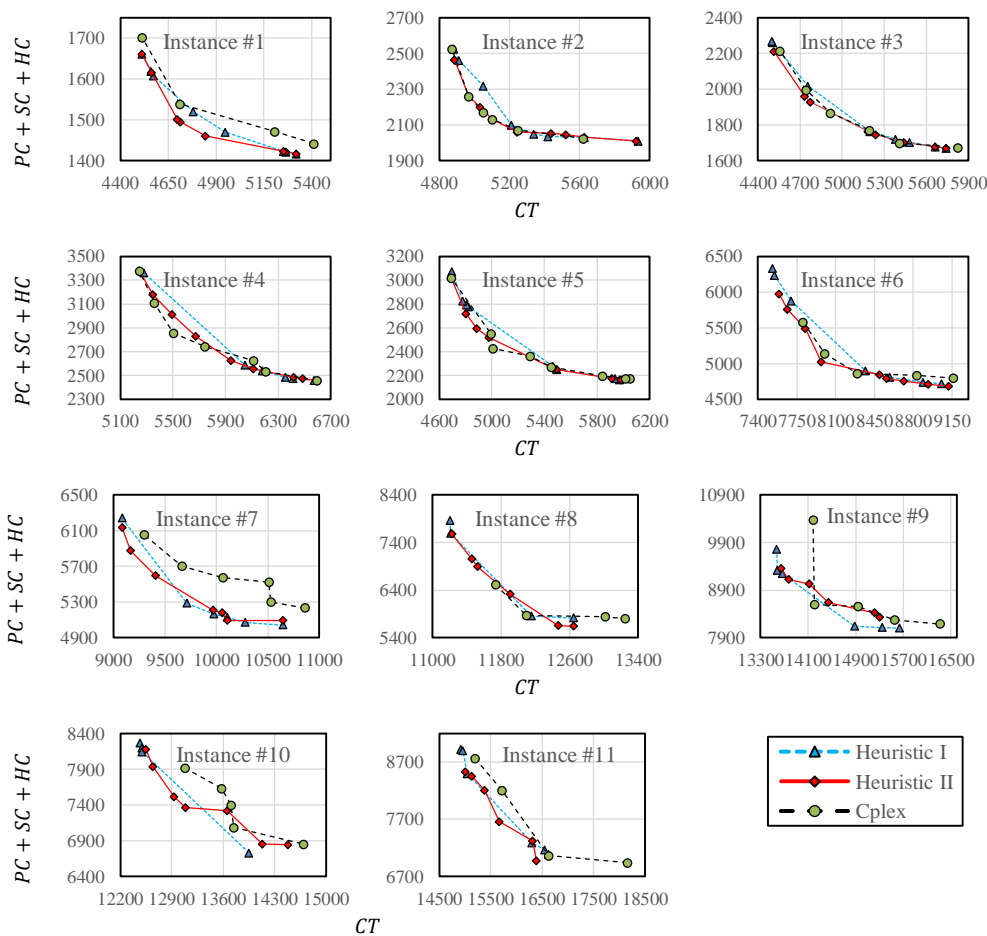


**Figure 3. Pareto frontier obtained by each method for the generated instances**

## 5. Conclusion

In this paper, an integrated cell formation and operation scheduling problem was presented. Multiple process routings, reentrant parts, and job splitting were the main issues that were addressed along with the proposed problem. The problem

_____

was formulated as a mixed-integer program so as to minimize the sum of processing, setup, and inter-cell handling costs, plus the total weighted flow time. Two heuristics were presented in order to find efficient solutions for medium- and large-sized instances. Eleven instances were generated at random and solved by the heuristics, and then, the results were compared to the solution derived from Cplex. The comparison results indicated that the proposed heuristics are able to produce better solutions compared to the Cplex, in a relatively shorter computational time. Also, it was concluded that it is better to implement both heuristics in order to achieve better solutions.

## REFERENCES

[1] **Arkat, J., Farahani, M.H., Hosseini, L. (2012),** *Integrating Cell Formation with Cellular Layout and Operations Scheduling*. *International Journal of Advanced Manufacturing Technology*, 61, 637–647;

[2] **Arkat, J., Farahani, M.H., Ahmadizar, F. (2012),** *Multi-objective Genetic Algorithm for Cell Formation Problem Considering Cellular Layout and Operations Scheduling*. *International Journal of Computer Integrated Manufacturing*, 25, 625–635;

[3] **Arkat, J., Ghahve, H. (2014),** *Scheduling of Virtual Manufacturing Cells with Outsourcing Allowed*. *International Journal of Computer Integrated Manufacturing*, 27, 1079–1089;

[4] **Boutsinas, B. (2013),** *Machine-part Cell Formation Using Biclustering*. *European Journal of Operational Research*, 230, 563–572;

[5] **Elmi, A., Solimanpur, M., Topaloglu, S., Elmi A. (2011),** *A Simulated Annealing Algorithm for the Job Shop Cell Scheduling Problem with Intercellular Moves and Reentrant Parts*. *Computers and Industrial Engineering*, 61, 171–178;

[6] **Ghezavati, V., Saidi-Mehrabad M. (2010),** Designing *Integrated Cellular Manufacturing Systems with Scheduling Considering Stochastic Processing Time*. *International Journal of Advanced Manufacturing Technology*, 48, 701–717;

[7] **Halat, K., Bashirzadeh, R. (2015),** Concurrent *Scheduling of Manufacturing Cells Considering Sequence-Dependent Family Setup Times and Intercellular Transportation Times*. *International Journal of Advanced Manufacturing Technology*, 77, 1907–1915;

[8] **Kesen, S.E., Güngör, Z. (2012),** Job *Scheduling in Virtual Manufacturing Cells with Lot-Streaming Strategy: A New Mathematical Model Formulation And A Genetic Algorithm Approach*. *The Journal of the Operational Research Society*, 63, 683–695;

[9]  **Li, D., Meng, X., Li, M., Tian, Y. (2016), An ACO-based** *Intercell Scheduling Approach for Job Shop Cells with Multiple Single Processing Machines and One Batch Processing Machine*. *Journal of Intelligent Manufacturing*, 27, 283–296;

[10] **Liu, C., Wang, J., Leung, J.Y.T., Li, K. (2016),** *Solving Cell Formation and Task Scheduling in Cellular Manufacturing System By Discrete Bacteria Foraging Algorithm*. *International Journal of Production Research*, 54, 923–944;

[11] **Rafiei, H., Rabbani, M., Gholizadeh, H., Dashti, H. (2016),** *A Novel Hybrid SA/GA Algorithm for Solving an Integrated Cell Formation–Job Scheduling Problem with Sequence-Dependent Set-Up Times*. *International Journal of Management Science and Engineering Management*, 11, 134–142;

[12] **Singh, N. (1993),** *Design of Cellular Manufacturing Systems: An Invited Review*. *European Journal of Operational Research*, 69, 284–291;

[13] **Solimanpur, M., Elmi, A. (2013),** *A Tabu Search Approach for Cell Scheduling Problem with Makespan Criterion*. *International Journal of Production Economics*, 141, 639–645;

[14] **Solimanpur, M., Vrat, P., Shankar, R. (2004),** *A Heuristic to Minimize Makespan of Cell Scheduling Problem*. *International Journal of Production Economics*, 88, 231–241;

[15] **Tavakkoli-Moghaddam, R., Javadian, N., Khorrami, A., Gholipour-Kanani, Y. (2010),** *Design of a Scatter Search Method for a Novel Multi-Criteria Group Scheduling Problem in a Cellular Manufacturing System*. *Expert Systems with Applications* 37, 2661–2669;

[16] **Wang, X., Tang, J., Yung, K.L. (2010),** *A Scatter Search Approach with Dispatching Rules for a Joint Decision of Cell Formation and Parts Scheduling in Batches*. *International Journal of Production Research*, 48, 3513–3534;

[17] **Wemmerlöv, U., Hyer, N.L. (1989),** *Cellular Manufacturing in the U.S. Industry: A Survey of Users*. *International Journal of Production Research*, 27, 1511–1530;

[18] **Wu, X., Chu, C.H., Wang, Y., Yue, D. (2007),** *Genetic Algorithms for Integrating Cell Formation with Machine Layout and Scheduling*. *Computers and Industrial Engineering*, 53, 277–289;

[19] **Yang, W.H., Liao, C.J. (1996),** *Group Scheduling on Two Cells with Intercell Movement*. *Computers & Operations Research*, 23, 997–1006;

[20] **Zeng, C., Tang, J., Yan, C. (2014),** *Job-Shop Cell-Scheduling Problem with Inter-Cell Moves and Automated Guided Vehicles*. *Journal of Intelligent Manufacturing*, 26, 845–859.