



HAL
open science

Three parallel task assignment problems with shared resources

Ali Diabat, Alexandre Dolgui, Wladyslaw Janiak, Mikhail Kovalyov

► To cite this version:

Ali Diabat, Alexandre Dolgui, Wladyslaw Janiak, Mikhail Kovalyov. Three parallel task assignment problems with shared resources. *IISE Transactions*, 2020, 52 (4), pp.478-485. 10.1080/24725854.2019.1680907. hal-02437934

HAL Id: hal-02437934

<https://hal.science/hal-02437934v1>

Submitted on 14 Jan 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Three manpower allocation problems in labor-intensive cellular manufacturing

Ali Diabat¹, Alexandre Dolgui², Wladyslaw Janiak³, Mikhail Y. Kovalyov⁴

¹ *New York University - Abu Dhabi, P.O. Box 129188, Abu Dhabi, UAE,
E-mail: diabat@nyu.edu*

² *IMT Atlantique, LS2N - UMR CNRS 6004, La Chantrerie, 4, rue Alfred Kastler - B.P.
20722, F-44307 Nantes Cedex 3, France, E-mail: alexandre.dolgui@imt-atlantique.fr*

³ *General Tadeusz Kościuszko Military University of Land Forces, Faculty of Management,
Wrocław, Poland, E-mail: wladyslaw.janiak@awl.edu.pl*

⁴ *United Institute of Informatics Problems, National Academy of Sciences of Belarus, Minsk,
Belarus, E-mail: kovalyov_my@newman.bas-net.by*

Abstract

We study three manpower allocation problems in labor-intensive cellular manufacturing, for which mathematical programming formulations were already developed and computer tested but no computational complexity analysis is provided in the literature. We present very efficient new solution procedures for two of these problems and prove that the third problem is NP-hard in the strong sense and that it can be solved efficiently for special cases. Solutions of the studied problems can be used for solving more general manpower allocation and cell loading problems in labor-intensive cellular manufacturing.

Keywords: Cellular manufacturing, Manpower allocation, Computational complexity, Polynomial algorithms.

1 Introduction

Cellular manufacturing is a practical realization of the group technology philosophy, whose scientific principles were first mentioned by Flanders [22] in 1925 and further developed and advocated by many scholars, some references in the chronological order are: Sokolovsky [45], Mitrofanov [36], Opitz [37], Burbidge [15], Ham et al. [27], Kusiak [34], Wemmerlöv and Hyer [55, 56], Harhalakis et al. [25], Heragu [26], Brandon [10], Askin et al. [2], Askin and Zhou [3], Tanaev et al. [53], Irani [29], Inman and Helm [28], Dolgui and Proth [20], Askin [4], Stevenson [46], Brusco [11], Wahyudin et al. [54], and many others. Group technology aims at reducing setup times and costs and increasing productivity, product quality and worker and customer satisfaction by clustering and matching products requiring the same or similar

operations and machines performing these operations. Manufacturing cell is a collection of machines and workers or robots operating these machines in a close neighborhood.

Many industries such as automotive industry, aircraft part manufacturing, garment production, jewelry manufacturing, medical device manufacturing utilize labor-intensive manufacturing cells. Each cell is usually assigned a team of full-skilled workers each of which can perform any operation. In labor-intensive cellular manufacturing, manpower is the major resource, and its optimal allocation to operations in a cell is one of the main problems. Various aspects of the workforce assignment problems in cellular and team-based manufacturing have been studied by King [33], Wild [57], Peters and Waterman [39], Wilson [58], Chen and Saxena [17], Johnson [31], Johnson and Johnson [32], Peterson [40], Russel et al. [43], Wirth et al. [59], Süer [47], Bukchin et al. [12], Süer and Bera [50], Akturk and Wilson [1], Davis and Mabert [18], Groover [24], Li et al. [35], Bukchin and Masin [13], Süer and Dagli [51], Cesani and Steudel [16], Bidanda et al. [8], Dimitriadis [19], Becker and Scholl [6, 7], Süer et al. [48, 49], Süer et al. [52], Battaïa and Dolgui [5], Rubin and Bai [42], Otto and Battaïa [38]. Besides, team-based approach is widely used to improve functioning of organizations (Rami Shani et al. [41], Janiak [30]).

Süer et al. [48, 49] consider assignment of full-skilled workers to cells intended for parallel execution of tasks in product manufacturing. A set of eligible assignment configurations (product, cell, number of workers) is given. The objectives are minimizing the number of tardy products and minimizing the total number of workers. The decision includes the number of cells to open, cell sizes in terms of the number of workers, product-cell assignment and product sequences for each cell. Süer et al. [52] extend these models by considering the minimization of the total product tardiness. The solution procedure in Süer et al. [52] is a two-stage one. In the first stage, a given number of workers is allocated over a given number of parallel tasks for a given single product type in a single cell. Three problems are formulated, which differentiate by the assumptions of whether a worker can be shared between several tasks or not, or whether worker sharing between the tasks is limited. Mathematical programming formulations are suggested and computer tested for these problems. The next solution stage addresses cell loading with alternative configurations determined by the number of workers and the product type for each cell.

We study computational complexity of the three problems of the first solution stage in

Süer et al. [52]. The problems are formulated in Section 2. The criterion is the same for all three problems. In [52], it is to maximize the minimum output rate over all tasks. In our paper, it is to minimize the maximum task time. Let t be the task processing time if it is assigned a single worker. If it is assigned y workers, then its processing time is t/y and the task output rate is $60y/t$ task cycles per hour if t is in minutes. It is easy to see that the criteria of maximizing the output rate and minimizing the maximum task time are equivalent. We have made our choice because min-max problems are more popular than max-min problems in the optimization literature. Section 3 contains properties of optimal solutions for two of the problems and a generalization of one of them, and low order polynomial time algorithms based on these properties. The third problem is proved NP-hard in the strong sense and several its special cases are shown to be polynomially solvable in Section 4. The paper concludes with a short summary of the results and suggestions for future research.

2 Problem formulations

The common part of all three studied problems is the following. There are s tasks to be performed by n full-skilled workers. The tasks are performed in parallel. Each worker has the same capacity, which we assume, without loss of generality, to be equal to one. In the general case, capacity of the same worker can be shared between the tasks, and several workers can be assigned to the same task. The processing time of task j is equal to t_j/y_j , where y_j is the total worker capacity assigned to it, $j = 1, \dots, s$. Denote by δ_{ij} a portion of the capacity of worker i assigned to task j , $0 \leq \delta_{ij} \leq 1$. We have $y_j = \sum_{i=1}^n \delta_{ij}$, $j = 1, \dots, s$. The problem is to distribute worker capacities between s tasks so that each task is assigned a non-zero capacity portion and the maximum task processing time, $\max_{j=1, \dots, s} \{t_j/y_j\}$, is minimized.

Three variations of the common problem are denoted as CN, DS and RS(u), which abbreviate “continuous”, “discrete” and “restricted up to u ”, respectively. In the problems CN and DS, variables y_j are real and integer numbers, respectively, which address the cases where the same worker can be shared and, respectively, it cannot be shared between several tasks. In the problem RS(u), variables y_j are real numbers with an additional requirement that the total worker capacity $\sum_{j=1}^s y_j$ can be distributed between the workers so that the same worker is assigned to at most u tasks for a given number u , $u \in \{1, \dots, s\}$. The input

size of any of the three problem variants is $O(s)$. Mathematical programming formulations of the three problems are given below. Introduce variable vector y with entries y_j and variable matrix δ with entries δ_{ij} , $i = 1, \dots, n$, $j = 1, \dots, s$. We call pair (y, δ) a *solution*.

Problem CN (with unrestricted worker sharing):

$$\min_y \max_{j=1, \dots, s} \left\{ \frac{t_j}{y_j} \right\}, \quad (1)$$

subject to

$$\sum_{j=1}^s y_j \leq n, \quad (2)$$

$$y_j > 0, \quad j = 1, \dots, s. \quad (3)$$

Denote optimal solution and optimal solution value for the problem CN as $(y^{(cn)}, \delta^{(cn)})$ and $F^{(cn)}$, respectively. Given vector $y^{(cn)}$, corresponding matrix $\delta^{(cn)}$ can be determined by the following *Wrap Around Rule*.

Wrap Around Rule.

Step 1. Define *integer task capacity deadlines* $D_j = \lfloor y_j^{(cn)} \rfloor$, $j = 1, \dots, s$. Assign workers $1, \dots, D_1$ in their maximum capacities to task 1: $\delta_{i1}^{(cn)} = 1$, $i = 1, \dots, D_1$. Assign workers $D_1 + 1, \dots, D_1 + D_2$ in their maximum capacities to task 2: $\delta_{i2}^{(cn)} = 1$, $i = D_1 + 1, \dots, D_1 + D_2$. Continue in the same fashion until assigning workers $\sum_{i=1}^{s-1} D_i, \dots, \sum_{i=1}^s D_i$ in their maximum capacities to task s : $\delta_{is}^{(cn)} = 1$, $i = \sum_{i=1}^{s-1} D_i, \dots, \sum_{i=1}^s D_i$. If $y_j^{(cn)}$, $j = 1, \dots, s$, are integer, then stop: $\delta^{(cn)}$ is determined. Otherwise, perform Step 2.

Step 2. Define *fractional task capacity deadlines* $d_j := y_j^{(cn)} - \lfloor y_j^{(cn)} \rfloor$, $j = 1, \dots, s$. Define set $S = \{j \mid d_j > 0\}$. To facilitate presentation, assume $S = \{1, \dots, s^0\}$. Determine task index j_1 such that $\sum_{j=1}^{j_1-1} d_j < 1 \leq \sum_{j=1}^{j_1} d_j$. Assign worker $w = \sum_{i=1}^s D_i + 1$ to tasks $1, \dots, j_1$ so that $\delta_{w,j}^{(cn)} = d_j$, $j = 1, \dots, j_1 - 1$ and $\delta_{w,j_1}^{(cn)} = 1 - \sum_{j=1}^{j_1-1} d_j$. If $\sum_{j=1}^{j_1} d_j > 1$, then re-set $S := \{j_1, j_1 + 1, \dots, s^0\}$ and $d_{j_1} := \sum_{j=1}^{j_1} d_j - 1$. If $\sum_{j=1}^{j_1} d_j = 1$, then re-set $S := \{j_1 + 1, j_1 + 2, \dots, s^0\}$. To facilitate presentation, assume $S = \{j_1, j_1 + 1, \dots, s^0\}$. Determine task index j_2 such that $\sum_{j=j_1}^{j_2-1} d_j < 1 \leq \sum_{j=j_1}^{j_2} d_j$. Assign worker $w + 1$ to tasks $j_1, j_1 + 1, \dots, j_2$ so that $\delta_{w+1,j}^{(cn)} = d_j$, $j = j_1, j_1 + 1, \dots, j_2 - 1$ and $\delta_{w+1,j_2}^{(cn)} = 1 - \sum_{j=j_1}^{j_2-1} d_j$. If $\sum_{j=j_1}^{j_2} d_j > 1$, then re-set $S := \{j_2, j_2 + 1, \dots, s^0\}$ and $d_{j_2} := \sum_{j=j_1}^{j_2} d_j - 1$. If $\sum_{j=j_1}^{j_2} d_j = 1$, then re-set $S := \{j_2 + 1, j_2 + 2, \dots, s^0\}$. Continue in the same fashion until assigning worker $w + k = \lceil \sum_{j=1}^s y_j^{(cn)} \rceil$ to tasks $j_k, j_k + 1, \dots, s^0$. \square

The Wrap Around Rule can be implemented to run in $O(s)$ time.

Problem DS (without worker sharing): the only difference with the problem CN is that constraints (3) are replaced with

$$y_j \in \{1, \dots, n\}, \quad j = 1, \dots, s. \quad (4)$$

Due to the constraints (4), problem DS has no solution if $n \leq s - 1$. If $n = s$, then there exists a unique feasible (and optimal) solution $y = (1, \dots, 1)$. Assume that $n \geq s + 1$ for the problem DS. Denote optimal solution and optimal solution value for the problem DS as $(y^{(ds)}, \delta^{(ds)})$ and $F^{(ds)}$, respectively. Given vector $y^{(ds)}$, corresponding matrix $\delta^{(ds)}$ can be determined by Step 1 of the Wrap Around Rule, in which notation “ cn ” is replaced by “ ds ”.

Introduce 0-1 variables z_{ij} such that $z_{ij} = 1$ if $\delta_{ij} > 0$. Note that we allow $z_{ij} = 1$ if $\delta_{ij} = 0$. Values δ_{ij} and z_{ij} , $j = 1, \dots, s$, $i = 1, \dots, n$, are variables in the problem with restricted worker sharing.

Problem RS(u) (with restricted worker sharing):

$$\min_{\delta} \max_{j=1, \dots, s} \left\{ \frac{t_j}{\sum_{i=1}^n \delta_{ij}} \right\}, \quad (5)$$

subject to

$$\sum_{j=1}^s z_{ij} \leq u, \quad i = 1, \dots, n, \quad (6)$$

$$\sum_{j=1}^s \delta_{ij} = 1, \quad i = 1, \dots, n, \quad (7)$$

$$z_{ij} \geq \delta_{ij}, \quad i = 1, \dots, n, \quad j = 1, \dots, s, \quad (8)$$

$$\delta_{ij} \geq 0, z_{ij} \in \{0, 1\}, \quad i = 1, \dots, n, \quad j = 1, \dots, s. \quad (9)$$

Constraints (6) guarantee that each worker is assigned to at most u tasks. Constraints (7) ensure that each worker is employed in its maximum capacity (equal to 1). Constraints (7) and (9) imply $\delta_{ij} \leq 1$ for all i and j . Constraints (8), (9) and relation $\delta_{ij} \leq 1$ imply $z_{ij} = 1$ if $\delta_{ij} > 0$. Denote optimal variable vector y , optimal variable matrix δ and optimal solution value for the problem RS(u) as $y^{(rsu)}$, $\delta^{(rsu)}$ and $F^{(rsu)}$, respectively.

Constraints (6)-(9) have the same effect as those in the model 1c in Süer et al. [52]. Note that the number of variables and constraints in the problem (5)-(9) is $O(sn)$, which is not polynomial because of n .

Equivalent formulations of the problems CN, DS and $RS(u)$ will be obtained if we reverse dividend and divisor in the objective function, replace min-max criterion with max-min one and allow $y_j, j = 1, \dots, s$, to take zero values. By doing so, the problems CN, DS and $RS(u)$ become a linear programming (LP) problem, an integer linear programming (ILP) problem and a mixed integer linear programming (MILP) problem, respectively. Such formulations are given by Sürer et al. [52]. While LP can be solved in polynomial time in the number of variables and constraints, ILP and MILP cannot be handled in polynomial time in their general form, unless $\mathcal{P} = \mathcal{NP}$. We shall demonstrate that the problems CN and DS can be solved in $O(s)$ and $O(s^2)$ time, respectively, and that the problem $RS(u)$ is NP-hard in the strong sense. These results improve solution approaches for the first two problems in [52], and justify the MILP approach for the third problem.

3 Properties and efficient solutions of problems CN and DS

We start with proving useful properties of an optimal solution of the problem CN.

Statement 1 *There exists an optimal vector $y^{(cn)}$ which satisfies the following properties:*

$$(i-cn) \sum_{j=1}^s y_j^{(cn)} = n,$$

$$(ii-cn) F^{(cn)} = (\sum_{j=1}^s t_j)/n,$$

$$(iii-cn) y_j^{(cn)} = t_j/F^{(cn)} = t_j n / (\sum_{j=1}^s t_j), j = 1, \dots, s.$$

Proof: Consider an optimal vector $y^{(cn)}$ and assume that property (i-cn) is not satisfied for it. Increase arbitrary values $y_j^{(cn)}$ so that this property is satisfied. Since this update does not increase optimal value $F^{(cn)}$, the new solution is optimal.

Now consider an optimal vector $y^{(cn)}$ for which property (i-cn) is satisfied. By the definition of the objective function, relations $t_j \leq y_j^{(cn)} F^{(cn)}$, $j = 1, \dots, s$, are satisfied. By summing left-hand sides and right-hand sides of these relations and taking into account $\sum_{j=1}^s y_j^{(cn)} = n$, we obtain $\sum_{j=1}^s t_j \leq nF^{(cn)}$ and $F^{(cn)} \geq (\sum_{j=1}^s t_j)/n$. Thus, $(\sum_{j=1}^s t_j)/n$ is a lower bound for $F^{(cn)}$.

Define $y_j = \frac{t_j n}{\sum_{r=1}^s t_r}$, $j = 1, \dots, s$. We have $\sum_{j=1}^s y_j = n$. Therefore, y is feasible for the problem CN. Moreover, $t_j/y_j = (\sum_{r=1}^s t_r)/n$, $j = 1, \dots, s$, which implies that y is optimal. Properties (i-cn), (ii-cn) and (iii-cn) are satisfied for it. \square

Statement 1 and the fact that the Wrap Around Rule runs in $O(s)$ time implies that the problem CN can be solved in $O(s)$ time.

We now pass to considering the problem DS. First note that if $\lfloor y_j^{(cn)} \rfloor = y_j^{(cn)}$, $j = 1, \dots, s$, then $y^{(cn)}$ is feasible for the problem DS, and therefore, $y^{(ds)} = y^{(cn)}$ and $F^{(ds)} = F^{(cn)}$. Assume that $\lfloor y_j^{(cn)} \rfloor < y_j^{(cn)}$ for some j , $1 \leq j \leq s$. In this case, $\sum_{j=1}^s \lfloor y_j^{(cn)} \rfloor < \sum_{j=1}^s y_j^{(cn)} = n$.

Theorem 1 *If $\sum_{j=1}^s \lfloor y_j^{(cn)} \rfloor < n$, then $F^{(ds)}$ belongs to the set $T := \left\{ \frac{t_j}{y_j} \mid y_j = \lfloor y_j^{(cn)} \rfloor - 1, \lfloor y_j^{(cn)} \rfloor - 2, \dots, \max\{1, \lfloor y_j^{(cn)} \rfloor - s + 1\}, j = 1, \dots, s \right\}$.*

Proof: By the definition of the value $F^{(ds)}$, this value is equal to t_j/y for some $j = 1, \dots, s$ and $y = 1, \dots, n$. Since $F^{(cn)}$ is a lower bound for $F^{(ds)}$ and $\sum_{j=1}^s \lfloor y_j^{(cn)} \rfloor < n$, we know that $F^{(ds)} \geq \min_{j=1, \dots, s} \left\{ \frac{t_j}{\lfloor y_j^{(cn)} \rfloor - 1} \right\}$. To prove the theorem, it remains to show that $F^{(ds)} \leq \max_{j=1, \dots, s} \left\{ \frac{t_j}{\max\{1, \lfloor y_j^{(cn)} \rfloor - s + 1\}} \right\}$.

Consider vector $y^{(0)} = \left(\max\{1, \lfloor y_1^{(cn)} \rfloor - s + 1\}, \dots, \max\{1, \lfloor y_s^{(cn)} \rfloor - s + 1\} \right)$. Introduce sets $J_1 = \{j \mid y_j^{(0)} = 1\}$ and $J_2 = \{j \mid y_j^{(0)} \geq 2\}$. We have $|J_1| + |J_2| = s$ and

$$\sum_{j=1}^s y_j^{(0)} = |J_1| + \sum_{j \in J_2} \lfloor y_j^{(cn)} \rfloor - |J_2|(s-1).$$

If $J_2 = \emptyset$, then, taking into account $n \geq s + 1$, obtain $\sum_{j=1}^s y_j^{(0)} = |J_1| = s < n$. If $J_2 \neq \emptyset$, then $|J_1| \leq s - 1$, $|J_2| \geq 1$ and

$$\sum_{j=1}^s y_j^{(0)} = |J_1| - |J_2|(s-1) + \sum_{j \in J_2} \lfloor y_j^{(cn)} \rfloor \leq \sum_{j \in J_2} \lfloor y_j^{(cn)} \rfloor \leq \sum_{j=1}^s \lfloor y_j^{(cn)} \rfloor < n.$$

Thus, $\sum_{j=1}^s y_j^{(0)} < n$. Consider another vector $y^{(1)}$ with $y_j^{(1)} = y_j^{(0)}$ for $j = 1, \dots, s-1$, and $y_s^{(1)} = n - \sum_{j=1}^{s-1} y_j^{(0)} > y_s^{(0)}$. It is clear that $\sum_{j=1}^s y_j^{(1)} = n$, i.e., $y^{(1)}$ is feasible, and

$$F^{(ds)} \leq \max_{j=1, \dots, s} \left\{ \frac{t_j}{y_j^{(1)}} \right\} \leq \max_{j=1, \dots, s} \left\{ \frac{t_j}{y_j^{(0)}} \right\} = \max_{j=1, \dots, s} \left\{ \frac{t_j}{\max\{1, \lfloor y_j^{(cn)} \rfloor - s + 1\}} \right\},$$

which completes the proof. \square

Consider $F \in T$ and a vector y satisfying $y_j \in \{0, 1, \dots, n\}$ and $\frac{t_j}{y_j} \leq F$, $j = 1, \dots, s$. Relations $\frac{t_j}{y_j} \leq F$ and integrality of y_j imply $y_j \geq \left\lceil \frac{t_j}{F} \right\rceil$, $j = 1, \dots, s$. Define vector

$$y^{(F)} = \left(\left\lceil \frac{t_1}{F} \right\rceil, \dots, \left\lceil \frac{t_s}{F} \right\rceil \right).$$

If $\sum_{j=1}^s \left\lceil \frac{t_j}{F} \right\rceil \leq n$, then $y^{(F)}$ is feasible for DS. Otherwise, if $\sum_{j=1}^s \left\lceil \frac{t_j}{F} \right\rceil > n$, then there is no vector y simultaneously satisfying conditions (2), (4) and relations $\frac{t_j}{y_j} \leq F$, $j = 1, \dots, s$, that

is, $F < F^{(ds)}$. We deduce that $F^{(ds)}$ is equal to the minimal value $F \in T$, for which relation $\sum_{j=1}^s \lceil \frac{t_j}{F} \rceil \leq n$ is satisfied. The value of $F^{(ds)}$ and the corresponding optimal vector $y^{(ds)}$ can be found by the following Bisection Search Procedure over the elements of the set T .

Bisection Search Procedure.

Input: Set T defined in Theorem 1. We stress that T is defined as a set and not as a multi-set, that is, all its elements are distinct.

Step 1. Select smallest element, denoted as A , in the set T . If $n \leq \sum_{j=1}^s \lceil \frac{t_j}{A} \rceil$, then $F^{(ds)} = A$ and $y^{(ds)} = y^{(A)}$. Stop. If $n > \sum_{j=1}^s \lceil \frac{t_j}{A} \rceil$, then remove A from T , select largest element, denoted as B , in T , determine feasible solution $y^{(B)}$ with $y_j^{(B)} = \lfloor t_j/B \rfloor$, $j = 1, \dots, s-1$, $y_s^{(B)} = n - \sum_{r=1}^{s-1} y_r^{(B)}$, and perform Step 2. Step 1 can be implemented to run in $O(s)$ time.

Step 2. If $|T| = 1$, then stop: $F^{(ds)} = B$ and $y^{(ds)} = y^{(B)}$. If $|T| \geq 2$, then select *median* M in the set T , which is the $\lceil |T|/2 \rceil$ -th smallest element in this set. This can be done in $O(|T|)$ time by employing the median finding technique of Blum et al. [9]. If $n \leq \sum_{j=1}^s \lceil \frac{t_j}{M} \rceil$, then update set T by removing elements larger than M from it, re-set $B := M$ and repeat Step 2. If $n > \sum_{j=1}^s \lceil \frac{t_j}{M} \rceil$, then update set T by removing M and elements smaller than M from it, and repeat Step 2. Both removals can be done in $O(s + |T|)$ time. If T^0 is the set T after any of the removals, then $|T^0| \leq |T|/2 + 1$. □

Since the cardinality of the set T is decreased almost twice in each iteration of Step 2 and the number of iterations of this step is $O(\log s)$, the Bisection Search Procedure can be implemented to run in $O(s \log s + |T| + |T|/2 + |T|/4 + |T|/8 + \dots + 1) = O(s \log s + |T|)$ time. Therefore, problem DS can be solved in $O(s^2)$ time.

The results for the problem CN can be extended for the following more general problem, which we denote as CN-POWER. Let α be a given constant, $\alpha > 0$. CN-POWER differs from CN only in that the processing time of task j is equal to $t_j/(y_j)^\alpha$ if y_j is the total worker capacity assigned to it, $j = 1, \dots, s$. We denote optimal solution and optimal solution value of the problem CN-POWER as $(y^{(po)}, \delta^{(po)})$ and $F^{(po)}$, respectively. If $0 < \alpha < 1$, then this problem is appropriate in situations when the per worker efficiency drops as the team size increases.

Statement 2 *There exists an optimal vector $y^{(po)}$ which satisfies the following properties:*

$$(i-po) \sum_{j=1}^s y_j^{(po)} = n,$$

$$\begin{aligned}
(ii-po) \quad & \frac{t_1}{(y_1^{(po)})^\alpha} = \dots = \frac{t_s}{(y_s^{(po)})^\alpha}, \\
(iii-po) \quad & F^{(po)} = \left(\frac{\sum_{j=1}^s (t_j)^{\frac{1}{\alpha}}}{n} \right)^\alpha, \\
(iv-po) \quad & y_j^{(po)} = \frac{n(t_j)^{\frac{1}{\alpha}}}{\sum_{j=1}^s (t_j)^{1/\alpha}}, \quad j = 1, \dots, s.
\end{aligned}$$

Proof: Consider an optimal vector $y^{(po)}$ and assume that property (i-po) is not satisfied for it. Increase arbitrary values $y_j^{(po)}$ so that this property is satisfied. Since this update does not increase the optimal value $F^{(po)}$, the new solution is optimal.

Consider an optimal vector $y^{(po)}$ for which property (i-po) is satisfied. Re-number tasks such that $\frac{t_1}{(y_1^{(po)})^\alpha} \leq \dots \leq \frac{t_s}{(y_s^{(po)})^\alpha}$. Assume that property (ii-po) is not satisfied. Then $\frac{t_1}{(y_1^{(po)})^\alpha} < \frac{t_s}{(y_s^{(po)})^\alpha}$. Determine δ such that $0 < \delta < y_1^{(po)}$ and $\frac{t_1}{(y_1^{(po)} - \delta)^\alpha} = \frac{t_s}{(y_s^{(po)} + \delta)^\alpha}$. The latter equality is equivalent to $\delta = \frac{\left(y_1^{(po)} \left(\frac{t_s}{t_1} \right)^{\frac{1}{\alpha}} \right) - y_s^{(po)}}{1 + \left(\frac{t_s}{t_1} \right)^{\frac{1}{\alpha}}}$. The inequality $\frac{\left(y_1^{(po)} \left(\frac{t_s}{t_1} \right)^{\frac{1}{\alpha}} \right) - y_s^{(po)}}{1 + \left(\frac{t_s}{t_1} \right)^{\frac{1}{\alpha}}} > 0$ is equivalent to the valid inequality $\frac{t_1}{(y_1^{(po)})^\alpha} < \frac{t_s}{(y_s^{(po)})^\alpha}$, and the inequality $\frac{\left(y_1^{(po)} \left(\frac{t_s}{t_1} \right)^{\frac{1}{\alpha}} \right) - y_s^{(po)}}{1 + \left(\frac{t_s}{t_1} \right)^{\frac{1}{\alpha}}} < y_1^{(po)}$ is equivalent to the valid inequality $-y_s^{(po)} < y_1^{(po)}$.

Consider a new solution by re-setting $y_1^{(po)} := y_1^{(po)} - \delta$ and $y_s^{(po)} := y_s^{(po)} + \delta$. The new solution is optimal, it satisfies property (i-po), and the number of distinct values $\frac{t_j}{(y_j^{(po)})^\alpha}$ reduces by one. Continue the described modification until these values become equal, in which case properties (i-po) and (ii-po) are satisfied.

Consider an optimal vector $y^{(po)}$ for which properties (i-po) and (ii-po) are satisfied. Property (ii-po) implies $\frac{t_j}{(y_j^{(po)})^\alpha} = F^{(po)}$, or equivalently,

$$y_j^{(po)} = \left(\frac{t_j}{F^{(po)}} \right)^{\frac{1}{\alpha}}, \quad j = 1, \dots, s. \quad (10)$$

By summing left-hand sides and right-hand sides of (10), we obtain $n = \sum_{j=1}^s \left(\frac{t_j}{F^{(po)}} \right)^{\frac{1}{\alpha}}$ and further $F^{(po)} = \left(\frac{\sum_{j=1}^s (t_j)^{\frac{1}{\alpha}}}{n} \right)^\alpha$, that is, property (iii-po) is satisfied. Property (iv-po) follows from (10) and property (iii-po). \square

Taking into account the fact that the Wrap Around Rule can be employed to convert $y^{(po)}$ into $\delta^{(po)}$, we deduce that the problem CN-POWER is solvable in $O(s)$ time under the assumption that any power function in Statement 2 is computable in $O(1)$ time. Computational complexity of the discrete analog of the problem CN-POWER is an open question.

4 Properties and strong NP-hardness of problem $\text{RS}(u)$

Similar to the problems CN and DS, it can be easily shown that there exists an optimal vector $y^{(rsu)}$, for which $\sum_{j=1}^s y_j^{(rsu)} = n$. It follows that the problem $\text{RS}(1)$ reduces to the problem DS and the problem $\text{RS}(s)$ reduces to the problem CN. In the rest of this section we assume that $2 \leq u \leq s - 1$. Let us prove more properties of the problem $\text{RS}(u)$, which throw light on its complexity.

Statement 3 *If $un < s$, then the problem $\text{RS}(u)$ has no solution.*

Proof: Since each worker can be assigned to at most u tasks, the maximum number of tasks assigned to n workers does not exceed un . Hence, if $un < s$, then at least one task will always remain unassigned. \square

Assume that the tasks are re-numbered such that $t_1 \geq \dots \geq t_s$.

Statement 4 *For the same instance of the problems CN and $\text{RS}(u)$, $2 \leq u \leq s - 1$, if $y_{s-1}^{(cn)} = t_{s-1}n / (\sum_{j=1}^s t_j) \geq 1$, then $y^{(rsu)} = y^{(cn)}$ and optimal matrix $\delta^{(rsu)}$, in which each worker is assigned to at most two tasks, can be obtained in $O(s)$ time from $y^{(cn)}$.*

Proof: Consider the same instance of the problems $\text{RS}(u)$, $2 \leq u \leq s - 1$, and CN, and optimal vector $y^{(cn)}$ with $y_{s-1}^{(cn)} \geq 1$. Determine $\delta^{(rsu)}$ as follows. Let I and J be the sets of integer and non-integer values of $y_j^{(cn)}$, respectively. From $y_{s-1}^{(cn)} \geq 1$ it follows that set J contains at most one value which is less than one. Calculate $w = \sum_{j \in I} y_j^{(cn)}$. Assign workers $1, \dots, w$ to tasks $j \in I$, one worker for each task.

To facilitate the proof, assume without loss of generality that $J = \{1, \dots, |J|\}$ and $y_1^{(cn)} = \min\{y_j^{(cn)} \mid j \in J\}$. Then $y_j^{(cn)} > 1$, $j = 2, 3, \dots, |J|$.

Perform the following computations for $h = 1, \dots, |J|$. Calculate $w_h = \lfloor y_h^{(cn)} \rfloor$. If $w_h \geq 1$, then assign workers $w+1, w+2, \dots, w+w_h$ to task h from J and re-set $w := w + w_h$. Calculate $\gamma_h = y_h^{(cn)} - w_h$. If $\gamma_h > 0$ and $h = |J|$, then assign worker $w+1$ to task h in the capacity γ_h and stop. If $\gamma_h > 0$ and $h \leq |J| - 1$, then assign worker $w+1$ to tasks h and $h+1$ from J in the capacities γ_h and $1 - \gamma_h$, respectively. Re-set $y_{h+1}^{(cn)} := y_{h+1}^{(cn)} - (1 - \gamma_h)$. Since the original value of $y_{h+1}^{(cn)}$ was greater than one, we have $y_{h+1}^{(cn)} > 0$ for the new value. Repeat the described computations for $h := h + 1$. Finally, all $n = \sum_{j=1}^s y_j^{(cn)}$ workers will be assigned and each of

them will be assigned to at most two tasks. For the given $y^{(cn)}$, the described computations require $O(s)$ time. \square

Let us show that conversion of the optimal vector $y^{(cn)}$ of an instance of the problem CN into an optimal solution of the same instance of the problem RS(u) is not always possible.

Statement 5 *There exists an instance of the problem RS(2) which has a solution such that $y^{(rs2)} \neq y^{(cn)}$ and $F^{(rs2)} > F^{(cn)}$, where $y^{(cn)}$ is an optimal vector of the same instance of the problem CN and $F^{(cn)}$ is the optimal value of CN.*

Proof: Consider the same instance of the problems RS(2) and CN, in which $n = 5$, $s = 8$, $t_1 = \dots = t_6 = 8$ and $t_7 = t_8 = 1$. According to the Statement 1, $F^{(cn)} = (\sum_{j=1}^8 t_j)/n = 10$, $y_1^{(cn)} = \dots = y_6^{(cn)} = (8 \cdot 5)/50 = 0.8$ and $y_7^{(cn)} = y_8^{(cn)} = (1 \cdot 5)/50 = 0.1$. If $0 < y_8^{(rs2)} \leq 0.1$, then at least one worker is assigned to task 8 in the capacity which does not exceed 0.1. Since $u = 2$ in the problem RS(2), this worker must be assigned to one of the tasks $1, \dots, 7$ in the capacity of at least 0.9, which cannot be realized if $y^{(rs2)} = y^{(cn)}$. Hence, $y^{(rs2)} \neq y^{(cn)}$. As $y^{(cn)}$ is the unique optimal vector for the problem CN and this problem is less restricted than RS2, $F^{(rs2)} > F^{(cn)}$. \square

We now prove that the problem RS(3) is difficult.

Theorem 2 *The problem RS(3) is NP-hard in the strong sense.*

Proof: We will use a reduction from the NP-hard in the strong sense problem 3-PARTITION, see Garey and Johnson [23].

3-PARTITION: Given $3m + 1$ positive integer numbers h_1, \dots, h_{3m} and H satisfying $\sum_{j=1}^{3m} h_j = mH$, does there exist a partition of the set $\{1, \dots, 3m\}$ into subsets X_1, \dots, X_m such that $\sum_{j \in X_i} h_j = H$ for $i = 1, \dots, m$?

Given an instance of 3-PARTITION, we construct an instance of the problem RS, in which $s = 3m$, $t_j = h_j$, $j = 1, \dots, s$, $n = m$ and $u = 3$. We show that the instance of 3-PARTITION has a solution if and only if the corresponding instance of RS3 has a feasible solution with value $F^{(rs3)} \leq H$.

Part “if”. Assume that the constructed instance of RS3 has a feasible solution with value $F^{(rs3)} \leq H$. In this solution, a non-zero fraction of the capacity of each worker is assigned to

each of at most $u = 3$ tasks and each task is assigned at least one non-zero fraction of the worker capacity. Consider variables $z_{ij} \in \{0, 1\}$ such that $z_{ij} = 1$ if task j is assigned a non-zero fraction of the capacity of worker i , $j = 1, \dots, 3m$, $i = 1, \dots, m$. The above mentioned assignment requirements can be written as

$$\sum_{j=1}^{3m} z_{ij} \leq 3, \quad i = 1, \dots, m, \quad (11)$$

and

$$\sum_{i=1}^m z_{ij} \geq 1, \quad j = 1, \dots, 3m. \quad (12)$$

By separately summing both sides of the relations (11), we obtain $\sum_{i=1}^m \sum_{j=1}^{3m} z_{ij} \leq 3m$, and by summing both sides of the relations (12), we obtain $\sum_{i=1}^m \sum_{j=1}^{3m} z_{ij} \geq 3m$. Therefore, $\sum_{i=1}^m \sum_{j=1}^{3m} z_{ij} = 3m$.

If $\sum_{i=1}^m z_{ij} \geq 2$ for some j , then (12) implies $\sum_{i=1}^m \sum_{j=1}^{3m} z_{ij} \geq 3m + 1$, which is a contradiction. Similarly, if $\sum_{j=1}^{3m} z_{ij} \leq 2$ for some i , then (11) implies $\sum_{i=1}^m \sum_{j=1}^{3m} z_{ij} \leq 3m - 1$, a contradiction again. We deduce $\sum_{i=1}^m z_{ij} = 1$, $j = 1, \dots, 3m$, and $\sum_{j=1}^{3m} z_{ij} = 3$, $i = 1, \dots, m$, which verbally means that each worker is assigned to exactly three tasks and no two workers share the same task. Denote by X_i the set of tasks assigned to worker i , $i = 1, \dots, m$. We showed that X_1, \dots, X_m is a partition of the the set $\{1, \dots, 3m\}$. Furthermore, $F^{(rs3)} \leq H$ implies

$$\frac{t_j}{\delta_{ij}} \leq H, \quad \text{for } j \in X_i, \quad i = 1, \dots, m, \quad (13)$$

where δ_{ij} is the fraction of the capacity of worker i assigned to task j , $\sum_{j \in X_i} \delta_{ij} = 1$, $i = 1, \dots, m$. By separately summing both sides of the relations (13) over $j \in X_i$, we obtain $\sum_{j \in X_i} t_j \leq H \left(\sum_{j \in X_i} \delta_{ij} \right) = H$, $i = 1, \dots, m$, which, together with $\sum_{j=1}^{3m} t_j = mH$, implies $\sum_{j \in X_i} h_j = \sum_{j \in X_i} t_j = H$ for $i = 1, \dots, m$. Hence, X_1, \dots, X_m is a solution of 3-PARTITION.

Part “only if”. Assume that X_1, \dots, X_m is a solution of an instance of 3-PARTITION. For the corresponding instance of RS3, construct a solution, in which fraction $y_{ij} = \frac{t_j}{H} = \frac{h_j}{H}$ of the capacity of worker i is assigned to task $j \in X_i$, $\sum_{j \in X_i} y_{ij} = (\sum_{j \in X_i} h_j)/H = 1$, $i = 1, \dots, m$. For this solution, each worker is assigned to exactly $u = 3$ tasks, each task is assigned exactly one worker, and

$$F^{(rs3)} = \max_{j \in X_i, i=1, \dots, m} \left\{ \frac{t_j}{y_{ij}} \right\} = H, \quad i = 1, \dots, m,$$

that is, X_1, \dots, X_m is a solution of the corresponding instance of RS. \square

Note that the instance of the problem RS(3) constructed in the above proof might not be labor-intensive because the number of workers n is three times smaller than the number of operations. The proof can easily be modified to show that the labor-intensive variant of the problem RS(3) with the number of workers much larger than the number of operations is NP-hard in the strong sense. Let $f(m)$ be any given polynomial function of m with positive integer values, for example, $f(m) = m^3$. To address the labor-intensive case, we update the instance of the problem RS(3) in Theorem 1 so that $s = 3m + 1$, $t_{3m+1} = f(m)H$ and $n = m + f(m)$. The other parameters remain unchanged. The only significant change in the proof is an observation that $F^{(rs3)} \leq H$ implies that the task $3m + 1$ is assigned the total capacity of at least $f(m)$.

5 Conclusions and extensions

We developed $O(s)$, $O(s)$ and $O(s^2)$ time algorithms for the problems CN, CN-POWER and DS, respectively, and proved strong NP-hardness of the problem RS(3). The problem RS(u) has no solution if $un < s$, and it can be solved in $O(s)$ time if the second smallest value $t_j n / (\sum_{j=1}^s t_j)$, $j = 1, \dots, s$, is at least one. The problem RS(1) reduces to the problem DS and the problem RS(s) reduces to the problem CN.

The following open questions are interesting. Does there exist an algorithm for the problem DS with the running time better than $O(s^2)$? Can the results for the problem CN-POWER be extended for the discrete analog of this problem? Can an optimal vector $y^{(cn)}$ be converted in polynomial time into an optimal solution $(y^{(rsu)}, \delta^{(rsu)})$ for the cases other than that in Statement 4?

The development of alternative MILP formulations and efficient optimal and approximation algorithms for the problem RS(u) are the perspective directions for future research. We also consider opportunities for translation of the optimization results on workforce management in manufacturing to workforce management in organizations.

References

- [1] Akturk, M.S., Wilson, G.R. (1998) A hierarchical model for the cell loading problem of cellular manufacturing systems. *International Journal of Production Research* 36(7),

2005-2023.

- [2] Askin, R.G., Selim, H.M., Vakharia, A.J. (1997) A methodology for designing flexible cellular manufacturing systems, *IIE Transactions* 29(7), 599-610.
- [3] Askin, R.G., Zhou, M. (1998) Formation of independent flow-line cells based on operation requirements and machine capabilities, *IIE Transactions* 30(4), 319-329.
- [4] Askin, R.G. (2013) Contributions to the design and analysis of cellular manufacturing systems, *International Journal of Production Research* 51(23-24), 6778-6787.
- [5] Battaïa, C., Dolgui, A. (2013) A taxonomy of line balancing problems and their solution approaches. *International Journal of Production Economics* 142(2), 259-277.
- [6] Becker, C., Scholl, A. (2006) A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research* 168, 694-715.
- [7] Becker, C., Scholl, A. (2009) Balancing assembly lines with variable parallel workplaces: Problem definition and effective solution procedure. *European Journal of Operational Research* 199 (2009), 359-374.
- [8] Bidanda, B., Ariyawongrat, P., Needy, K.L., Norman, B.A., Tharmmaphornphilas, W. (2005) Human related issues in manufacturing cell design, implementation, and operation: a review and survey. *Computers & Industrial Engineering* 48, 507-523.
- [9] Blum, M., Floyd, R.W., Pratt, V.R., Rivest, R.L., Tarjan, R.E. (1973) Time bounds for selection. *Journal of Computer and System Sciences* 7 (4), 448-461.
- [10] Brandon, J. (1996) *Cellular manufacturing: Integrating technology and management*. Research Studies Press, Somerset, England.
- [11] Brusco, M.J. (2015) An exact algorithm for maximizing grouping efficacy in part-machine clustering. *IIE Transactions* 47 (6), 653-671.
- [12] Bukchin, J., Dar-El, E.M., Rubinovitz, J. (1997) Team oriented assembly system design: A new approach. *International Journal of Production Economics* 51, 47-57.

- [13] Bukchin, J., Masin, M. (2004) Multi-objective design of team oriented assembly systems. *European Journal of Operational Research* 156, 326-352.
- [14] Bullen, P.S. (2003) *Handbook of Means and Their Inequalities*, vol. 560. Kluwer Academic Publishers, Dordrecht, The Netherlands.
- [15] Burbidge, J.L. (1975) *The Introduction to group technology*. John Wiley, New York.
- [16] Cesani, V.I., Steudel, H.J. (2005) A study of labor assignment flexibility in cellular manufacturing systems. *Computers & Industrial Engineering* 48(3), 571-591.
- [17] Chen, J., Saxena, U. (1987) A technical note: Optimal manpower assignment to injection molding machines. *Industrial Management* 29(6), 31-33.
- [18] Davis, D.J., Mabert, V.A. (2000) Order dispatching and labor assignment in cellular manufacturing systems. *Decision Sciences* 31(4), 745-771.
- [19] Dimitriadis, S.G. (2006) Assembly line balancing and group working: A heuristic procedure for workers groups operating on the same product and workstation. *Computers & Operations Research* 33(5), 2757-2774.
- [20] Dolgui, A., Proth, J.M. (2010) *Supply chain engineering: Useful methods and techniques*, chapter 10: Manufacturing layout. Springer, Berlin Heidelberg New York.
- [21] Fetter, R.N. (1955) The assignment of workers to service automatic machines. *The Journal of Industrial Engineering* VI(5), 22-30.
- [22] Flanders, R.E. (1925) Design, manufacture, and production control of a standard machine. *Transactions of ASME* 26, 691-738.
- [23] Garey M.R., Johnson, D.S. (1979) *Computers and intractability: A guide to the theory of NP-completeness*. Freeman, San Francisco.
- [24] Groover, M. P. (2001) *Automation, production systems, and computer integrated manufacturing*. Prentice Hall, New Jersey, USA.

- [25] Harhalakis, G., Nagi, R., Proth, J-M. (1990) An efficient heuristic in manufacturing cell formation for group technology applications. *International Journal of Production Research* 28(1), 185-198.
- [26] Heragu, S.S. (1994) Group technology and cellular manufacturing. *IEEE Trans Syst Man Cybern* 24(2), 203-214.
- [27] Ham, I., Hitomi, K., Yoshida, T. (1985) *Group technology*. Kluwer-Nijhoff, Dordrecht.
- [28] Inman, R.A., Helm, M. (2006) *Encyclopedia of management*. Gale Cengage Learning, Detroit.
- [29] Irani, S. (1999) *Handbook of cellular manufacturing systems*. John Wiley & Sons, New York.
- [30] Janiak, W.A. (2017) *Kształtowanie struktur organizacyjnych. Rola i wpływ na bezpieczeństwo organizacji*. Wrocław: Wydawnictwo Akademii Wojsk Lądowych. (in Polish)
- [31] Johnson, R.V. (1991) Balancing assembly lines for team and work groups. *International Journal of Production Research* 29(6), 1205-1214.
- [32] Johnson, D., Johnson, F. (1991) *Joining together: Group theory and group skills*. Prentice Hall, Englewood Cliffs, NJ.
- [33] King, J.R. (1970) On the optimum size of workforce engaged in the servicing of automatic machines. *International Journal of Production Research* 8(3), 207-220.
- [34] Kusiak, A. (1987) The generalized group technology concept. *International Journal of Production Research* 25 (4), 561-569.
- [35] Li, X., Li, Y.Z., Lim, A., Rodrigues, B. (2003) Manpower allocation with time windows and job teaming constraints. *Proceedings - Annual Meeting of the Decision Sciences Institute*, pp. 1717-1722.
- [36] Mitrofanov, S.P. (1959) *The scientific principles of group technology*. Lenizdat, Leningrad. (In Russian)

- [37] Opitz, H. (1970) A classification system to describe workpieces: Part I and II. Pergamon, Oxford.
- [38] Otto, A., Battaia, O. (2017) Reducing physical ergonomic risks at assembly lines by line balancing and job rotation: A survey. *Computers & Industrial Engineering* 111, 467-480.
- [39] Peters, T., Waterman, R. (1982) *In search of excellence*. Harper and Row, New York.
- [40] Peterson, D. (1991) *Teamwork: New management ideas for the nineties*. Victor Gollancz, London.
- [41] Rami Shani, A.B., Mohrman, S.A., Pasmore, W.A., Stymne, B., Adler, N. (Eds.) (2007) *Handbook of collaborative management research*. Thousand Oaks, CA: Sage.
- [42] Rubin, P.A., Bai, L. (2015) Forming competitively balanced teams. *IIE Transactions* 47(6), 620-633.
- [43] Russell, R. S., Huang, P. Y., Leu, Y. (1991) A study of labor allocation strategies in cellular manufacturing. *Decision Sciences* 22(3), 594-611.
- [44] Shtub, A. (1984) The effect of incompleteness cost on line balancing with multiple manning of work stations. *International Journal of Production Research* 22, 235-245.
- [45] Sokolovsky A.P. (1939) *Foundations of machine building technology*. Mashgiz, Moscow. (In Russian)
- [46] Stevenson, W.J. (2014) *Operations management*, 12th edition. McGraw-Hill Education, New York.
- [47] Süer, G.A. (1996) Optimal worker assignment and cell loading in labor intensive manufacturing cells. *Computers and Industrial Engineering* 31 (12), 155-158.
- [48] Süer, G.A., Arıkan, F., Babayigit, C. (2008) Bi-objective cell loading problem with non-zero setup times with fuzzy aspiration levels in labour-intensive manufacturing cells. *International Journal of Production Research* 46 (2), 371-404.

- [49] Süer, G.A., Arıkan, F., Babayigit, C. (2009) Effects of different fuzzy workers on fuzzy bi-objective cell loading problem in labor-intensive manufacturing cells. *Computers and Industrial Engineering* 56, 476-488.
- [50] Süer, G.A., Bera, I.S. (1998) Multi period cell loading and cell size determination, *Computers and Industrial Engineering* 35 (1-2), 85-88.
- [51] Süer, G. A., Daglı, C. (2005) Intra-cell manpower transfers and cell loading in labor-intensive manufacturing cells. *Computers & Industrial Engineering* 48(3), 643-655.
- [52] Süer, G.A., Kamat, K., Mese, E., Huang, J. (2013) Minimizing total tardiness subject to manpower restriction in labor-intensive manufacturing cells. *Mathematical and Computer Modelling* 57, 741-753.
- [53] Tanaev, V.S., Kovalyov, M.Y., Shafransky, Y.M. (1998) *Scheduling theory. Group technologies*. Minsk, IEC NANB. (In Russian)
- [54] Wahyudin, R.S., Sutopo, W., Hisjam, M., Hardiono, R.S. (2016) Resource allocation model to find optimal allocation of workforce, material, and tools in an aircraft line maintenance. *Lecture Notes in Engineering and Computer Science* 2, 782-787.
- [55] Wemmerlöv, U., Hyer, N.L. (1987) Research issues in cellular manufacturing. *International Journal of Production Research* 25(3), 413-431.
- [56] Wemmerlöv, U., Hyer, N.L. (1989) Cellular manufacturing in the US industry: a survey of users. *International Journal of Production Research* 27(9), 1511-1530.
- [57] Wild, R. (1975) *Work organization: a study of manual work and mass production*. John Wiley & Sons, New Jersey, USA.
- [58] Wilson, J.M. (1986) Formulation of a problem involving assembly lines with multiple manning of work stations. *International Journal of Production Research* 24(1), 59-63.
- [59] Wirth, G. T., Mahmoodi, F., Mosier, C.T. (1993) An investigation of scheduling policies in a dual-constrained manufacturing cell. *Decision Sciences* 24(4), 761-788.