Computers & Industrial Engineering 154 (2021) 107172

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

# A novel algorithm of cell formation with alternative machines and multiple-operation-type machines

Ming-Liang Li *

*Department of Industrial Management, Oriental Institute of Technology, Banqiao Dist, New Taipei City, Taiwan*

ARTICLE INFO

*Keywords:*
Alternative machine
Operation type
Group technology
Group technology efficiency
Cellular manufacturing

ABSTRACT

Several different machines can often perform the same type of operation, so an operation can be performed on a part by selecting any of the machines offering that same function. Moreover, many machines are not limited to performing only one type of operation on a part, so a part can be fabricated using a single machine to perform more than one operation. Some papers have discussed alternative routings and machines, but have not considered a machine that can perform more than one operation on a part. The concept of using an alternative machine capable of performing several operation types was only introduced into group technology (GT) in 2014. In this paper, the major purpose is to improve traditional GT with considering alternative machines and multiple-operation-type machines. A novel algorithm is proposed. The result can recommend which machine can be used to provide which type of operation on a specific part. Two numerical examples are presented to compare the algorithm with others. Especially, there are sixty-four resolutions that could be suggested to the manager in numerical example one. The traditional GT algorithm cannot solve this problem. Occasionally, more than one result with the same GT efficiency can be provided.

## 1. Introduction

A factory leader must achieve the goals of increased productivity and decreased processing time and material wastage. The use of cellular manufacturing (CM) affords numerous advantages, such as a 45.6% decrease in throughput time, 39.3% decrease in material handling time, and it can reduce work-in-process inventory and setup time (Wemmerlov & Hyer, 1989). Thus, the two most fundamental benefits associated with CM are reductions in throughput time and inventory (Hyer & Wemmerlov, 2002). The optimum machine cell formation results in a reduction of the overall processing time and material handling costs (Hazarika & Laha, 2018; Mohammadi & Forghani, 2016). Moreover, "parts movement" wastes time and reduces assembly line efficiency. The concept of group technology (GT) has been implemented in CM to solve this problem (hereafter, the GT problem) (Shahdi-Pashaki, Teymourian, Kayvanfar, Komaki, & Sajadi, 2015; Zohrevand, Rafiei, & Zohrevand, 2016; Liu, Wang, & Leung, 2018; Nalluri, Kannan, & Gao, 2019). GT is a manufacturing philosophy and provides a methodology for sorting machines into machine cells and parts into part families (Boe & Cheng, 1991; Parkin & Li, 1997; Gupta, Devika, Valarmathi, Sowmiya, & Shinde, 2014; Danilovic & Ilic, 2019; Won, 2020; Li, 2020; Rahimi, Arkat, & Farughi, 2020). A GT environment can minimize the makespan (Qin, Zhang, & Bai, 2016; Zhang et al., 2018), reduce resource allocation costs (Liang et al., 2019), and find the optimal job schedule (Lu, Wang, Ji, & He, 2017; Miao, 2019). The benefits derived from the GT manufacturing system also include reductions of setup time, batch size and work-in-process inventory, and an improvement of quality (Singh & Rajamani, 1996). Numerous algorithms have been developed to solve the GT problem, such as heuristic, genetic, and neural network algorithms.

GT is based on the idea of using similarity to group parts (Suzic, 2012; Won & Logendran, 2015; Huang & Yan, 2019). A typical heuristic algorithm used in GT is the similarity coefficient method (McAuley, 1972):

$$s_{ij} = \frac{N_{ij}}{N_i + N_j - N_{ij}}$$

where $N_i$ and $N_j$ represent the number of parts processed by machines $i$ and $j$, respectively, and $N_{ij}$ represents the number of parts processed by both machines $i$ and $j$.

Kusiak and Cho (1992) improved the similarity coefficient method and considered bottleneck parts or bottleneck machines in the similarity coefficient algorithm while solving the GT problem. Kumar and Singh

* Address: 58, Sec. 2, Sze-Chuan Rd, Banqiao Dist, New Taipei City, Taiwan.
*E-mail address:* fg017@mail.oit.edu.tw.

https://doi.org/10.1016/j.cie.2021.107172

Available online 4 February 2021

**Nomenclature**

| | |
|---|---|
| $m$ | number of machines |
| $n$ | number of parts |
| $r$ | number of operation types |
| $m_i$ | $i^{th}$ machine |
| $p_i$ | $i^{th}$ part |
| $o_i$ | $i^{th}$ operation type |
| $A = \{a_{ij}\}_{m\times n}$ | incidence matrix |
| $B_{machine} = \{b_{ij}\}_{m\times m}$ | closeness matrix for machines |
| $B_{part} = \{b_{ij}\}_{n\times n}$ | closeness matrix for parts |
| $OP = \left\{op_{ij}\right\}_{r\times n}$ | operation type-part matrix |
| $MO = \{mo_{ij}\}_{m\times r}$ | machine-operation type matrix |
| $G_i$ | $i^{th}$ group |
| $g$ | number of groups |
| $S = \{s_{ij}\}_{gn}$ | group-part matrix |
| $\eta$ | group efficiency |
| $e_b$ | total number of 1's in the major blocks |
| $e_0$ | total number of 1's in the stray blocks |
| $Q_i$ | the number of machines in the $i^{th}$ cell |
| $P_i$ | the number of parts in the $i^{th}$ family |
| $q$ | weighting factor |
| $\upsilon_m$ | total number of times that machines move to different groups |
| $\upsilon_p$ | total number of times that parts move to different groups |

(2017) provided a novel similarity score-based two-phase heuristic approach to solve the dynamic cellular facility layout problem (DCFLP). The first step is to form a machine-cell based on similarity scores between machines.

Laha and Hazarika (2017) developed a heuristic approach based on the Euclidean distance matrix. This method differs from the similarity coefficient algorithm by using distance instead of similarity coefficients. The parts are sorted by considering the smallest Euclidean distance.

A genetic algorithm has also been proposed to solve the GT problem (Agustin-Blas et al., 2011; Boulif & Atif, 2006; Feng, Li, & Sethi, 2018; Hazarika & Laha, 2015; Imran, Kang, Lee, Jahanzaib, & Aziz, 2017). Using the genetic algorithm to solve the GT problem involves at least five steps: initial problem determination, fitness evaluation, population selection, crossover, and mutation (Hazarika & Laha, 2015). The genetic algorithm might find a solution, but it is usually not the best solution.

Although traditional algorithms are limited to solving two-dimensional GT problems, Parkin and Li (1997) attempted to solve GT problems of three or more dimensions by using a heuristic algorithm. The main concept of their algorithm is to consider all incidence matrices as independent and then to combine all closeness matrices. In addition, Bootaki, Mahdavi, and Paydar (2016) considered not only the part-machine incidence matrix, but also the worker-machine task matrix. They tried to minimize the voids of both worker-machine and worker-worker incidence matrices.

## 2. Problem description

The element in an incidence matrix of the traditional GT is either 1 or 0. The incidence matrix is defined as follows:

$A = \{a_{ij}\}$ (Boe & Cheng, 1991).

Here, $a_{ij} = \begin{cases} 1, & \text{if part } j \text{ visits machine } i \\ 0, & \text{otherwise} \end{cases}$

Consider the following example.

$$A = \begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \end{array} \begin{array}{c} \begin{array}{cccc} p_1 & p_2 & p_3 & p_4 \end{array} \\ \begin{bmatrix} 1 & & 1 & \\ & 1 & & 1 \\ 1 & & 1 & \\ & 1 & & 1 \end{bmatrix} \end{array}$$

where $a_{11} = 1$; this implies that means $p_1$ must be operated using $m_1$. Moreover, $a_{21} = 0$; this implies that $p_1$ should not be operated using $m_2$.

Some articles have discussed alternative routings (Hazarika & Laha, 2018; Mehdizadeh, Niaki, & Rahimi, 2016; Feng, Da, Xi, Pan, & Xia, 2017; Lian, Liu, Li, Evans, & Yin, 2014; Alhourani, 2016; Eguia, Molina, Lozano, & Racero, 2017; Forghani & Fatemi Ghomi, 2020; Rahimi et al., 2020), but papers have seldom mentioned alternative machines and operation types. It was not until 2014 that Navaei and ElMaraghy (2014) introduced the concept of using an alternate machine with several operation types into GT. In this paper, the operation types are also considered in an incidence matrix.

Suppose two operation types (such as drilling a hole and welding) have to be performed on part $\delta$; thus, the operation type-part matrix can be defined as follows:

$$OP = \begin{array}{r} \\ \textit{drilling a hole} \\ \textit{welding} \end{array} \begin{array}{c} \textit{part } \delta \\ \begin{bmatrix} 1 \\ 1 \end{bmatrix} \end{array}$$

Usually, more than one type of machine can be used on some parts to perform the same type of operation. In addition, a part can be fabricated using a single machine to perform more than one operation type. Thus, a single machine with multiple operation types must be considered. For example, a computer numerical control (CNC) milling machine can drill a hole and mill, as shown in Fig. 1.

Thus, the "machine-operation type" matrix can be defined as follows:

$$MO = \begin{array}{r} \\ \textit{milling machine} \end{array} \begin{array}{c} \begin{array}{cc} \textit{drilling a hole} & \textit{milling} \end{array} \\ \begin{bmatrix} 1 & \qquad 1 \end{bmatrix} \end{array}$$

In addition, having alternative machines capable of performing the same operation type means that the operation process (such as drilling a hole) could be completed by any one of them, as shown in Fig. 2.

Thus, in order to explain the question clearer, let's suppose an "operation type-part" matrix as follows:

$$OP = \begin{array}{c} \\ o_1 \\ o_2 \\ o_3 \\ o_4 \end{array} \begin{array}{c} \begin{array}{cccc} p_1 & p_2 & p_3 & p_4 \end{array} \\ \begin{bmatrix} 1 & & 1 & \\ & & & 1 \\ & & 1 & \\ & 1 & & 1 \end{bmatrix} \end{array}$$

where $op_{11} = 1$ implies that operation type $o_1$ has to be performed on $p_1$, and $op_{21} = 0$ implies that operation type $o_2$ does not have to be performed on $p_1$. Moreover, $o_1$, $o_2$, $o_3$, and $o_4$ represent drilling a hole, shearing, boring, and welding, respectively.

Suppose the "machine-operation type" matrix can be obtained as follows:

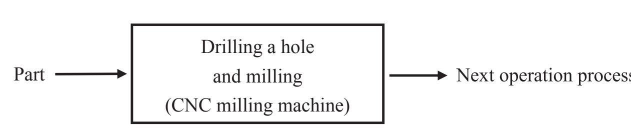


**Fig. 1.** A part is operated on by a single machine with multiple operation types.

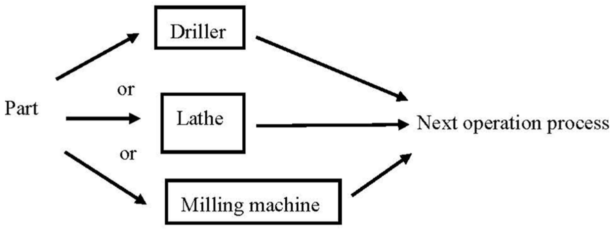


**Fig. 2.** A part is operated on by one of three types of machine.

$$MO = \begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \end{array} \begin{array}{c} \begin{array}{cccc} o_1 & o_2 & o_3 & o_4 \end{array} \\ \begin{bmatrix} 1 & & & \\ & 1 & & \\ 1 & & 1 & \\ & & & 1 \\ 1 & & 1 & \end{bmatrix} \end{array}$$

Here, when $mo_{11}$, $mo_{31}$, and $mo_{51}$ are equal to 1, it is implied that operation type $o_1$ could be performed by machine $m_1$, $m_3$, or $m_5$, respectively.

Thus, $p_1$ could be operated on by $m_1$, $m_3$, or $m_5$, as shown in Fig. 3.

The machine-operation type solutions for part $p_3$ are as shown in Fig. 4.

Traditional group technology usually provides only one answer. Although Navaei and ElMaraghy (2014) introduced the concept of an alternative machine with several operation types into GT, their method cannot form machine cells. Li and Parkin (2002) introduced the concept of "OR," which is the same concept as an alternative machine, but operation types were not considered.

In addition, traditional GT focuses on "1" or "0" in an incidence matrix, but an incidence matrix cannot reflect the problem presented. Thus, this paper proposes a novel algorithm that can recommend both an alternative machine and an operation type.

## 3. Methodology

The inputs of the algorithm are the operation type-part matrix and the machine-operation type matrix. The outputs are the groups of machines and parts. A flow chart of the algorithm proposed in this article is shown in Fig. 5.

### 3.1. Operation-part matrix

Consider that a factory comprises *m* machines, *n* parts, and *r* types of

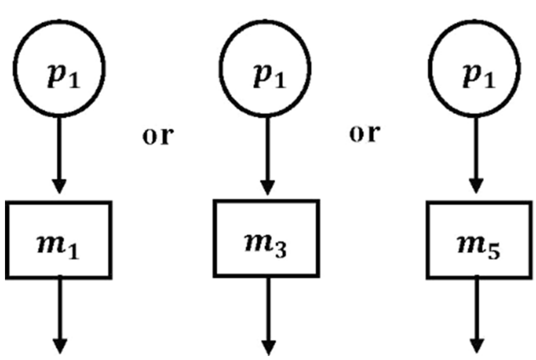


**Fig. 3.** $p_1$ could be operated on by one of three processes.

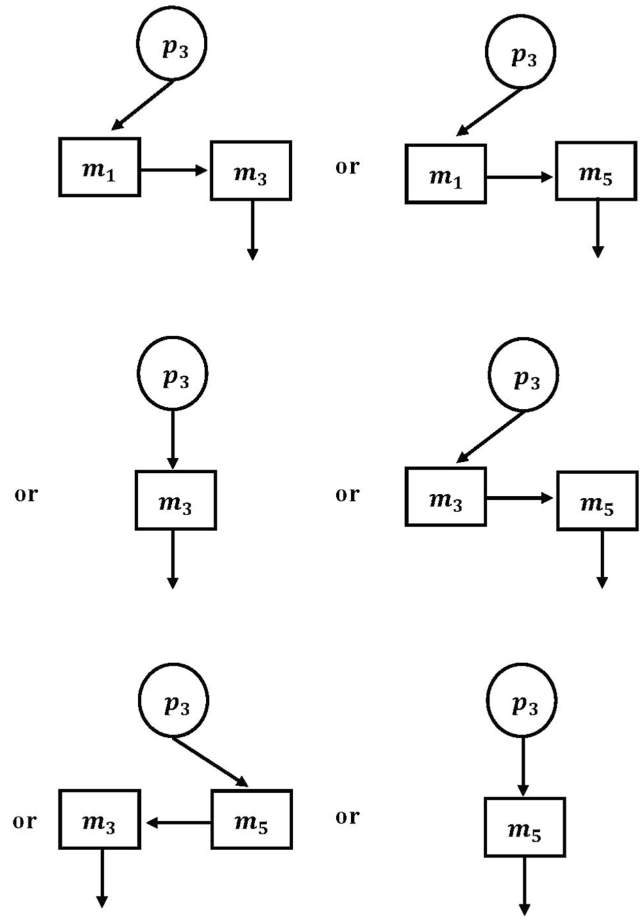


**Fig. 4.** $p_3$ could be operated on by one of six processes.

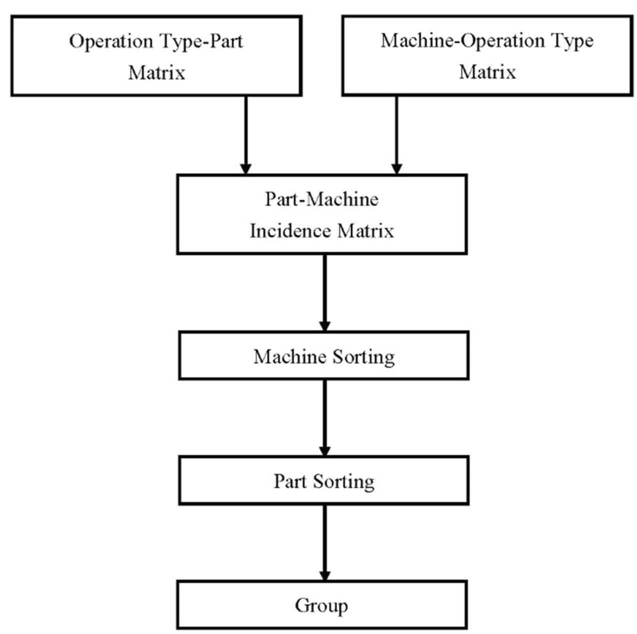


**Fig. 5.** A flow chart of the algorithm proposed in this article.

operations. Each part must undergo one or more types of operations. Thus, the relation between operation types and parts can be defined as follows:

$$OP = \{op_{ij}\}_{r\times n}$$

where $op_{ij} = \begin{cases} 1, \textit{if part j needs operation type i} \\ 0, \textit{otherwise} \end{cases}$

### 3.2. Machine-operation matrix

There is at least one machine that can perform a type of operation. Occasionally, an operation type may be performed by more than one machine. For this case, the machine–operation matrix can be defined as follows:

$$MO = \{mo_{ij}\}_{m\times r}$$

where $mo_{ij} = \begin{cases} 1, \textit{if operation type j may be done by machine i} \\ 0, \textit{otherwise} \end{cases}$

### 3.3. Incidence matrix

The incidence matrix of the traditional GT is defined as follows:

$$\mathrm{A} = \{a_{ij}\}_{m\times n}$$

where $a_{ij} = \begin{cases} 1, \textit{if machine i operates part j} \\ 0, \textit{otherwise} \end{cases}$

In this paper, the incidence matrix A must be modified and redefined as follows:

$$A = MO \times OP = \{mo_{ij}\}_{m\times r} \times \{op_{ij}\}_{r\times n} \tag{1}$$

where

$$a_{ij} = \sum_{k=1}^{r} mo_{ik} op_{kj}$$

The traditional $a_{ij}$ is equal to either 1 or 0 but is defined as follows in this paper:

$$a_{ij} = \{x | x = o_s, mo_{is} op_{sj} \neq 0, 1 \leq s \leq r\} \tag{2}$$

Moreover, $a_{ij}$ is a set, but the symbol {} is ignored in the incidence matrix.

### 3.4. Machine sorting

The closeness matrix of machine $B_{machine} = \{b_{ij}\}_{m\times m}$ is to determine the similarity between machines. Thus,

$$b_{pq} = \sum_{k=1}^{n} a_{pk} a_{qk}, p \neq q, b_{pp} = 0$$

where $a_{pk}a_{qk}$ is equal to 1 or 0 (Boe & Cheng, 1991).

However, $b_{ij}$ must be modified as follows:

$$b_{ij} = \sum_{k=1}^{n} a_{ik} a_{jk} \tag{3}$$

where

$$a_{ik}a_{jk} = \begin{cases} 0, \{[a_{ik} = a_{jk}] \wedge [n(a_{ik}) = n(a_{jk}) = 1]\} \vee \{min.[n(a_{ik}), n(a_{jk})] = 0\} \\ 1, otherwise \end{cases}$$

Forming a machine cell involves three steps:

Step 1: Machine grouping is initiated with the $m_i$ that has the highest number of relations.

Note that $m_i \in G_1$ and has the highest value of $\max.\{b_{uv}\}$, $1 \leq u$, $v \leq m$, in the $i^{th}$ row of $B_{machine}$. A tie is broken by selecting the smallest value of $i$.

Step 2: Find machine $m_k$ that has the highest value in $B_{machine}$, not including the entries of grouped machines' rows.

$$\begin{cases} m_k \in G_\xi, where\ (b_{kj} = max.\{b_{kt}\}, 1 \leq t \leq m) \bigwedge (m_j \in G_\xi) \\ \text{A tie is broken by choosing the smallest } j. \\ m_k \in a\ new\ group, otherwise \end{cases} \tag{4}$$

Step 3: Repeat step 2 until all machines are grouped.

### 3.5. Part sorting

First, machine cells must be formed. Subsequently, part families must be formed. Four steps are involved in assigning groups to parts to form part families.

Step 1: After machine groups are formed, the incidence matrix A should be arranged to A' based on the machine cells.

Step 2: A matrix $\mathrm{S} = \{s_{ij}\}_{gn}$ must be defined. There are $g$ rows and $n$ parts. The rows are $G_1$, $G_2$, …, and $G_g$ instead of machines.

where

$$s_{ij} = n(a'_{\alpha j} \cup a'_{\beta j} \cdots \cup a'_{\sigma j}) \tag{5}$$

$$where\ m_\alpha, m_\beta, \cdots, m_\sigma \in G_i$$

Step 3:

$$\text{Let } p_u \in G_v, \text{ where, } s_{vu} = max.(s_{iu}, i = 1, \cdots, g) \tag{6}$$

A tie is broken by selecting the highest efficiency.

Step 4: Repeat step 3 until all parts are grouped.

Two efficiency measurements have been proposed. One was defined by Chandrasekharan and Rajagopalan (1986) (Eq. (7)), and the other one was proposed by Li (2007) for two-dimensional GT (Eq. (8)).

$$\eta = q\eta_1 + (1-q)\eta_2 = q\frac{e_b}{\sum Q_i P_i} + (1-q)\left(1 - \frac{e_0}{mn - \sum Q_i P_i}\right) \tag{7}$$

$$\eta = \eta_1 \eta_2 = \left(1 - \frac{\sum Q_i P_i}{mn}\right)\left(1 - \frac{(\upsilon_m + \upsilon_p)}{(m+n) \times (g-1)}\right) \tag{8}$$

## 4. Numerical examples

This paper presents two numerical examples to demonstrate the operation of the proposed algorithm.

### 4.1. Example 1 and a comparison with Navaei and ElMaraghy's algorithm

Suppose that there is a factory with 5 machines, 6 parts, and 13 operation types.

$$
\mathrm{OP}= \begin{array}{c c}
 & \begin{array}{cccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \end{array} \\
\begin{array}{c} o_1 \\ o_2 \\ o_3 \\ o_4 \\ o_5 \\ o_6 \\ o_7 \\ o_8 \\ o_9 \\ o_{10} \\ o_{11} \\ o_{12} \\ o_{13} \end{array} &
\left[\begin{array}{cccccc}
1 & & & & & \\
1 & & & & & \\
 & 1 & & & & \\
 & 1 & & & & \\
 & & 1 & & & \\
 & & 1 & & & \\
 & & 1 & & & \\
 & & & 1 & & \\
 & & & 1 & & \\
 & & & & 1 & \\
 & & & & 1 & \\
 & & & & & 1 \\
 & & & & & 1
\end{array}\right]
\end{array}
$$

$$
\mathrm{MO}= \begin{array}{c c}
 & \begin{array}{ccccccccccccc} o_1 & o_2 & o_3 & o_4 & o_5 & o_6 & o_7 & o_8 & o_9 & o_{10} & o_{11} & o_{12} & o_{13} \end{array} \\
\begin{array}{c} m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \end{array} &
\left[\begin{array}{ccccccccccccc}
1 & & 1 & & 1 & 1 & & & & 1 & & 1 & \\
 & 1 & & & & 1 & 1 & & & & & 1 & \\
1 & & 1 & 1 & 1 & & & 1 & & 1 & & & \\
 & 1 & 1 & & 1 & & 1 & & & & & & 1 \\
 & & & & & & & & 1 & & 1 & &
\end{array}\right]
\end{array}
$$

Through Eqs. (1) and (2), for instance, $a_{m_2p_3}$ or $a_{23}$ can be obtained as follows:

$$
a_{23}= \begin{array}{c c}
 & \begin{array}{ccccccccccccc} o_1 & o_2 & o_3 & o_4 & o_5 & o_6 & o_7 & o_8 & o_9 & o_{10} & o_{11} & o_{12} & o_{13} \end{array} \\
m_2 & \left[\begin{array}{ccccccccccccc}
 & 1 & & & & 1 & 1 & & & & & 1 &
\end{array}\right]
\end{array}
\begin{array}{c c}
 & p_3 \\
\begin{array}{c} o_1 \\ o_2 \\ o_3 \\ o_4 \\ o_5 \\ o_6 \\ o_7 \\ o_8 \\ o_9 \\ o_{10} \\ o_{11} \\ o_{12} \\ o_{13} \end{array} &
\left[\begin{array}{c}
 \\ \\ \\ \\ 1 \\ 1 \\ 1 \\ \\ \\ \\ \\ \\ \\
\end{array}\right]
\end{array}
$$

$$= 0\times0+1\times0+0\times0+0\times0+0\times1+1\times1+1\times1+0\times0+0\times0+0\times0$$

$$+0\times0+1\times0+0\times0$$

$$= 0+0+0+0+0+1+1+0+0+0+0+0+0$$

There are two none zeros, noted as $\{o_6, o_7\}$, but the symbol {} is ignored in this article.

Thus, the incidence matrix A is defined as follows:

$$A = \begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \end{array} \begin{array}{c} \begin{array}{cccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \end{array} \\ \begin{bmatrix} o_1 & o_3 & o_5,o_6 & & o_{10} & o_{12} \\ o_2 & & o_6,o_7 & & & o_{12} \\ o_1 & o_3,o_4 & o_5 & o_8 & o_{10} & \\ o_2 & & o_5,o_7 & & & o_{13} \\ & o_4 & & o_9 & o_{11} & \end{bmatrix} \end{array}$$

Through Eq. (3), the entry $b_{m_1m_3}$ or $b_{13}$ of closeness matrix $B_{machine}$ is obtained as follows:

$$\begin{array}{c} \\ m_1 \\ m_3 \end{array} \begin{array}{c} \begin{array}{cccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \end{array} \\ \begin{bmatrix} o_1 & o_3 & o_5,o_6 & & o_{10} & o_{12} \\ o_1 & o_3,o_4 & o_5 & o_8 & o_{10} & \end{bmatrix} \\ \begin{array}{cccccc} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 0 & 1 & 1 & 0 & 0 & 0 \end{array} \end{array}$$

$$b_{13} = 0+1+1+0+0+0 = 2$$

Thus, $B_{machine}$ is obtained as follows:

$$B_{machine} = \begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \end{array} \begin{array}{c} \begin{array}{ccccc} m_1 & m_2 & m_3 & m_4 & m_5 \end{array} \\ \begin{bmatrix} - & 2 & 2 & 3 & 2 \\ 2 & - & 2 & 2 & 0 \\ 2 & 2 & - & 2 & 3 \\ 3 & 2 & 2 & - & 0 \\ 2 & 0 & 3 & 0 & - \end{bmatrix} \end{array}$$

In step 1, let $m_1$ belong to $G_1$ due to it having the highest number in $B_{machine}$.

In step 2, through Eq. (4), let $m_3$ belong to a new group, called $G_2$, due to the highest number in row $m_3$ being $b_{35}$, but $m_5$ does not belong to any group. Next, let $m_4$ belong to $G_1$ due to the highest number in row $m_4$ being $b_{41}$ and $m_1$ belonging to $G_1$. In the same way, two groups are formed: $G_1 = \{m_1, m_4, m_2\}$ and $G_2 = \{m_3, m_5\}$.

In part sorting, the first step is to rearrange incidence matrix $A$ to $A$'. Thus, the following is obtained:

$$A' = \begin{array}{cc} & \begin{matrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \end{matrix} \\ \begin{matrix} m_1 \\ m_4 \\ m_2 \\ m_3 \\ m_5 \end{matrix} & \begin{bmatrix} o_1 & o_3 & o_5,o_6 & & o_{10} & o_{12} \\ o_2 & & o_5,o_7 & & & o_{13} \\ o_2 & & o_6,o_7 & & & o_{12} \\ o_1 & o_3,o_4 & o_5 & o_8 & o_{10} & \\ & o_4 & & o_9 & o_{11} & \end{bmatrix} \end{array}$$

Using Eq. (5) yields the following:

$$S = \begin{array}{cc} & \begin{matrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \end{matrix} \\ \begin{matrix} G_1 \\ G_2 \end{matrix} & \begin{bmatrix} 2 & 1 & 3 & 0 & 1 & 2 \\ 1 & 2 & 1 & 2 & 2 & 0 \end{bmatrix} \end{array}$$

Applying Eq. (6) yields $p_1, p_3, p_6 \in G_1$ and $p_2, p_4, p_5 \in G_2$. Thus, the sorted incidence matrix A is as follows:

$$A_{sort} = \begin{array}{cc} & \begin{matrix} p_1 & p_3 & p_6 & p_2 & p_4 & p_5 \end{matrix} \\ \begin{matrix} m_1 \\ m_4 \\ m_2 \\ m_3 \\ m_5 \end{matrix} & \begin{bmatrix} o_1 & o_5,o_6 & o_{12} & o_3 & & o_{10} \\ o_2 & o_5,o_7 & o_{13} & & & \\ o_2 & o_6,o_7 & o_{12} & & & \\ o_1 & o_5 & & o_3,o_4 & o_8 & o_{10} \\ & & & o_4 & o_9 & o_{11} \end{bmatrix} \end{array}$$

Thus, $o_2$ of $p_1$ could be done by $m_4$ or $m_2$; $o_5$ of $p_3$ could be done by $m_1$ or $m_4$; $o_6$ of $p_3$ could be done by $m_1$ or $m_2$; $o_7$ of $p_3$ could be done by $m_4$ or $m_2$; $o_{12}$ of $p_6$ could be done by $m_1$ or $m_2$; $o_4$ of $p_2$ could be done by $m_3$ or $m_5$. There are sixty-four resolutions.

By conducting two measurements of GT efficiency, namely Chandrasekharan's measurement (with q $= 0.5$) and Li's measurement, and if the operation type or loading in each workstation is balanced, one resolution is suggested as follows:

$$A_{sort} = \begin{array}{cc} & \begin{array}{cccccc} p_1 & p_3 & p_6 & p_2 & p_4 & p_5 \end{array} \\ \begin{array}{c} m_1 \\ m_4 \\ m_2 \\ m_3 \\ m_5 \end{array} & \left[\begin{array}{cccccc} o_1 & o_5 & & & & \\ o_2 & o_7 & o_{13} & & & \\ & o_6 & o_{12} & & & \\ & & & o_3 & o_8 & o_{10} \\ & & & o_4 & o_9 & o_{11} \end{array}\right] \end{array}$$

Or rewritten as follows:

$$A_{sort} = \begin{array}{cc} & \begin{array}{cccccc} p_1 & p_3 & p_6 & p_2 & p_4 & p_5 \end{array} \\ \begin{array}{c} m_1 \\ m_4 \\ m_2 \\ m_3 \\ m_5 \end{array} & \left[\begin{array}{cccccc} 1 & 1 & & & & \\ 1 & 1 & 1 & & & \\ & 1 & 1 & & & \\ & & & 1 & 1 & 1 \\ & & & 1 & 1 & 1 \end{array}\right] \end{array}$$

Chandrasekharan's $\eta$ is equal to

$$\eta = 0.5 \times \frac{13}{3 \times 3 + 2 \times 3} + 0.5 \times \left(1 - \frac{0}{5 \times 6 - (3 \times 3 + 2 \times 3)}\right) = 0.933$$

Li's $\eta$ is equal to

$$\eta = \left(1 - \frac{(3 \times 3 + 2 \times 3)}{5 \times 6}\right)\left(1 - \frac{0}{(5 + 6) \times (2 - 1)}\right) = 0.5$$

By considering the higher GT efficiency, one resolution is suggested as follows in this article:

$$A_{sort} = \begin{array}{cc} & \begin{array}{cccccc} p_1 & p_3 & p_6 & p_2 & p_4 & p_5 \end{array} \\ \begin{array}{c} m_1 \\ m_4 \\ m_2 \\ m_3 \\ m_5 \end{array} & \left[\begin{array}{cccccc} o_1 & o_5, o_6 & o_{12} & & & \\ o_2 & o_7 & o_{13} & & & \\ & & & & & \\ & & & o_3 & o_8 & o_{10} \\ & & & o_4 & o_9 & o_{11} \end{array}\right] \end{array}$$

Or rewritten as follows:

$$
A_{sort} = \begin{array}{c} \\ m_1 \\ m_4 \\ m_2 \\ m_3 \\ m_5 \end{array}
\begin{array}{c}
\begin{array}{cccccc} p_1 & p_3 & p_6 & p_2 & p_4 & p_5 \end{array} \\
\left[\begin{array}{cccccc}
1 & 1 & 1 & & & \\
1 & 1 & 1 & & & \\
 & & & & & \\
 & & & 1 & 1 & 1 \\
 & & & 1 & 1 & 1
\end{array}\right]
\end{array}
$$

In this resolution, $m_2$ could be a spare machine, and the GT efficiency is as follows:

Chandrasekharan's $\eta$ is equal to

$$\eta = 0.5 \times \frac{12}{2 \times 3 + 2 \times 3} + 0.5 \times \left(1 - \frac{0}{5 \times 6 - (2 \times 3 + 2 \times 3)}\right) = 1$$

Li's $\eta$ is equal to

$$\eta = \left(1 - \frac{(2 \times 3 + 2 \times 3)}{5 \times 6}\right)\left(1 - \frac{0}{(5 + 6) \times (2 - 1)}\right) = 0.6$$

However, there is only one resolution for part families, and the machine cells have not been formed by Navaei and ElMaraghy (2014) algorithm. Their method only suggested $p_3, p_6, p_1, p_2 \in G_1$ and $p_4$, $p_5 \in G_2$, as shown in Fig. 6.

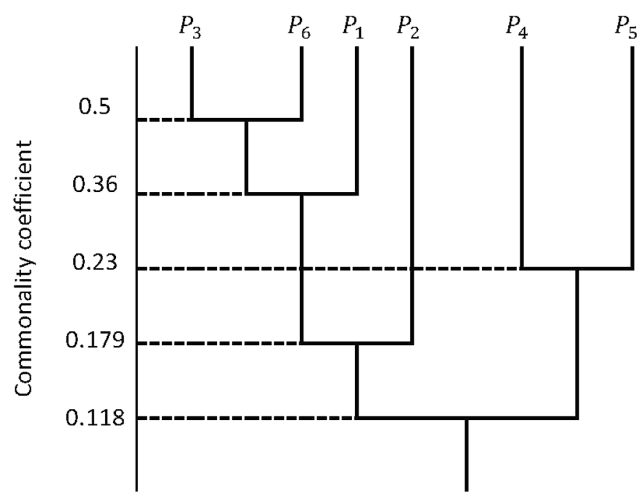


**Fig. 6.** Dendrogram presentation by Navaei and ElMaraghy's algorithm.

### 4.2. Example 2 and a comparison with Li and Parkin's algorithm

Another example was demonstrated in the study by Li and Parkin (2002), and it is described as follows:

$$
A = \begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \\ m_6 \\ m_7 \end{array}
\begin{array}{c}
\begin{array}{cccccccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 & p_8 & p_9 & p_{10} \end{array} \\
\left[\begin{array}{cccccccccc}
1 & & 1 & & & & 1 & & & \mathrm{C} \\
 & & & 1 & & 1 & & \mathrm{D} & & \mathrm{A} \\
1 & 1 & 1 & & & & 1 & 1 & & \mathrm{A} \\
 & & & & 1 & & & \mathrm{D} & 1 & \mathrm{A} \\
 & & & 1 & & 1 & & \mathrm{E} & & \mathrm{C} \\
1 & & 1 & & & & 1 & & & \mathrm{B} \\
 & & & & 1 & & & \mathrm{E} & 1 & \mathrm{B}
\end{array}\right]
\end{array}
$$

This example demonstrates the logical “OR” operation in GT. For example, $p_8$ must be operated by $m_3$, either $m_2$ or $m_4$, and either $m_5$ or $m_7$. Operation types were not considered in the study by Li and Parkin, which suggested only one resolution, as follows:

$$
\mathrm{A}_{sort} = \begin{array}{c} \\ m_3 \\ m_1 \\ m_6 \\ m_5 \\ m_2 \\ m_7 \\ m_4 \end{array}
\begin{array}{c}
\begin{array}{cccccccccc} p_{10} & p_1 & p_3 & p_7 & p_2 & p_8 & p_4 & p_6 & p_5 & p_9 \end{array} \\
\left[\begin{array}{cccccccccc}
\mathrm{A} & 1 & 1 & 1 & 1 & 1 & & & & \\
\mathrm{C} & 1 & 1 & 1 & & & & & & \\
\mathrm{B} & 1 & 1 & 1 & & & & & & \\
 & & & & & \mathrm{E} & 1 & 1 & & \\
 & & & & & \mathrm{D} & 1 & 1 & & \\
 & & & & & & & & 1 & 1 \\
 & & & & & & & & 1 & 1
\end{array}\right]
\end{array}
$$

Or rewritten as follows,

$$
\mathrm{A}_{sort} = \begin{array}{c} \\ m_3 \\ m_1 \\ m_6 \\ m_5 \\ m_2 \\ m_7 \\ m_4 \end{array}
\begin{array}{c}
\begin{array}{cccccccccc} p_{10} & p_1 & p_3 & p_7 & p_2 & p_8 & p_4 & p_6 & p_5 & p_9 \end{array} \\
\left[\begin{array}{cccccccccc}
1 & 1 & 1 & 1 & 1 & 1 & & & & \\
1 & 1 & 1 & 1 & & & & & & \\
1 & 1 & 1 & 1 & & & & & & \\
 & & & & & 1 & 1 & 1 & & \\
 & & & & & 1 & 1 & 1 & & \\
 & & & & & & & & 1 & 1 \\
 & & & & & & & & 1 & 1
\end{array}\right]
\end{array}
$$

However, this problem could be also solved using the method suggested in this paper. The problem-solving process can be initiated using the incidence matrix A as follows:

$$
A = \begin{array}{c} \\ m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \\ m_6 \\ m_7 \end{array}
\begin{array}{c}
\begin{array}{cccccccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 & p_7 & p_8 & p_9 & p_{10} \end{array} \\
\left[\begin{array}{cccccccccc}
o_1 & & o_5 & & & & o_{14} & & & o_{22} \\
 & & & o_8 & & o_{12} & & o_{17} & & o_{23} \\
o_2 & o_4 & o_6 & & & & o_{15} & o_{18} & & o_{23} \\
 & & & & o_{10} & & & o_{17} & o_{20} & o_{23} \\
 & & & o_9 & & o_{13} & & o_{19} & & o_{22} \\
o_3 & & o_7 & & & & o_{16} & & & o_{24} \\
 & & & & o_{11} & & & o_{19} & o_{21} & o_{24}
\end{array}\right]
\end{array}
$$

Solving this problem through the algorithm presented in this paper reveals three machine cells and three part families. The sorted incidence matrix is as follows.

$$
A_{sort} = \begin{array}{c} \\ m_1 \\ m_3 \\ m_6 \\ m_2 \\ m_5 \\ m_4 \\ m_7 \end{array}
\begin{array}{c}
\begin{array}{cccccccccc} p_1 & p_2 & p_3 & p_7 & p_{10} & p_4 & p_6 & p_8 & p_5 & p_9 \end{array} \\
\left[\begin{array}{cccccccccc}
o_1 & o_4 & o_5 & o_{14} & o_{22} & & & & & \\
o_2 & & o_6 & o_{15} & o_{23} & & & o_{18} & & \\
o_3 & & o_7 & o_{16} & o_{24} & & & & & \\
 & & & & o_{23} & o_8 & o_{12} & o_{17} & & \\
 & & & & o_{22} & o_9 & o_{13} & o_{19} & & \\
 & & & & o_{23} & & & o_{17} & o_{10} & o_{20} \\
 & & & & o_{24} & & & o_{19} & o_{11} & o_{21}
\end{array}\right]
\end{array}
$$

However, two resolutions are suggested because these two results have the same highest GT efficiency levels, as defined by Chandrasekharan and Li, shown in Table 1 and Table 2.

Chandrasekharan's $\eta$ is equal to

$$\eta = 0.5 \times \frac{23}{3\times5+2\times3+2\times2} + 0.5 \times \left(1 - \frac{1}{7\times10-25}\right) = 0.95$$

Li's $\eta$ is equal to

$$\eta = \left(1 - \frac{(3\times5+2\times3+2\times2)}{(7\times10)}\right)\left(1 - \frac{(1+1)}{(7+10)\times(3-1)}\right) = 0.61$$

The two sorted incidence matrices $A_{sort}$ are as follows:

**Table 2**
One resolution of example 2 and a comparison with Li and Parkin's algorithm.

| | Algorithm proposed in this study | Li and Parkin's algorithm |
|---|---|---|
| Machine cells | $\{m_1,m_3,m_6\}$, $\{m_2,m_5\}$, $\{m_4,m_7\}$ | $\{m_3,m_1,m_6\}$, $\{m_5,m_2\}$, $\{m_7,m_4\}$ |
| Part families | $\{p_1,p_2,p_3,p_7,p_{10}\}$, $\{p_4,p_6\}$, $\{p_8,p_5,p_9\}$ | $\{p_{10},p_1,p_3,p_7,p_2\}$, $\{p_8,p_4,p_6\}$,$\{p_5,p_9\}$ |
| Chandrasekharan's $\eta$, q = 0.5 | 0.95 | 0.95 |
| Li's $\eta$ | 0.61 | 0.61 |

$$
A_{sort} = \begin{array}{c} \\ m_1 \\ m_3 \\ m_6 \\ m_2 \\ m_5 \\ m_4 \\ m_7 \end{array}
\begin{array}{c}
\begin{array}{cccccccccc} p_1 & p_2 & p_3 & p_7 & p_{10} & p_4 & p_6 & p_8 & p_5 & p_9 \end{array} \\
\left[\begin{array}{cccccccccc}
o_1 & o_4 & o_5 & o_{14} & o_{22} & & & & & \\
o_2 & & o_6 & o_{15} & o_{23} & & & o_{18} & & \\
o_3 & & o_7 & o_{16} & o_{24} & & & & & \\
 & & & & & o_8 & o_{12} & o_{17} & & \\
 & & & & & o_9 & o_{13} & o_{19} & & \\
 & & & & & & & & o_{10} & o_{20} \\
 & & & & & & & & o_{11} & o_{21}
\end{array}\right]
\end{array}
$$

**Table 1**
One resolution of example 2 and a comparison with Li and Parkin's algorithm.

| | Algorithm proposed in this study | Li and Parkin's algorithm |
|---|---|---|
| Machine cells | $\{m_1,m_3,m_6\}$, $\{m_2,m_5\}$, $\{m_4,m_7\}$ | $\{m_3,m_1,m_6\}$, $\{m_5,m_2\}$, $\{m_7,m_4\}$ |
| Part families | $\{p_1,p_2,p_3,p_7,p_{10}\}$, $\{p_4,p_6,p_8\}$,$\{p_5,p_9\}$ | $\{p_{10},p_1,p_3,p_7,p_2\}$, $\{p_8,p_4,p_6\}$,$\{p_5,p_9\}$ |
| Chandrasekharan's $\eta$,q = 0.5 | 0.95 | 0.95 |
| Li's $\eta$ | 0.61 | 0.61 |

and

$$\mathrm{A}_{sort} = \begin{array}{cc} & \begin{array}{cccccccccc} p_1 & p_2 & p_3 & p_7 & p_{10} & p_4 & p_6 & p_8 & p_5 & p_9 \end{array} \\ \begin{array}{c} m_1 \\ m_3 \\ m_6 \\ m_2 \\ m_5 \\ m_4 \\ m_7 \end{array} & \left[\begin{array}{cccccccccc} o_1 & o_4 & o_5 & o_{14} & o_{22} & & & & & \\ o_2 & & o_6 & o_{15} & o_{23} & & & o_{18} & & \\ o_3 & & o_7 & o_{16} & o_{24} & & & & & \\ & & & & & o_8 & o_{12} & & & \\ & & & & & o_9 & o_{13} & & & \\ & & & & & & & o_{17} & o_{10} & o_{20} \\ & & & & & & & o_{19} & o_{11} & o_{21} \end{array}\right] \end{array}$$

Or rewritten as follows,

$$\mathrm{A}_{sort} = \begin{array}{cc} & \begin{array}{cccccccccc} p_1 & p_2 & p_3 & p_7 & p_{10} & p_4 & p_6 & p_8 & p_5 & p_9 \end{array} \\ \begin{array}{c} m_1 \\ m_3 \\ m_6 \\ m_2 \\ m_5 \\ m_4 \\ m_7 \end{array} & \left[\begin{array}{cccccccccc} 1 & 1 & 1 & 1 & 1 & & & & & \\ 1 & & 1 & 1 & 1 & & & 1 & & \\ 1 & & 1 & 1 & 1 & & & & & \\ & & & & & 1 & 1 & 1 & & \\ & & & & & 1 & 1 & 1 & & \\ & & & & & & & & 1 & 1 \\ & & & & & & & & 1 & 1 \end{array}\right] \end{array}$$

and

$$\mathrm{A}_{sort} = \begin{array}{cc} & \begin{array}{cccccccccc} p_1 & p_2 & p_3 & p_7 & p_{10} & p_4 & p_6 & p_8 & p_5 & p_9 \end{array} \\ \begin{array}{c} m_1 \\ m_3 \\ m_6 \\ m_2 \\ m_5 \\ m_4 \\ m_7 \end{array} & \left[\begin{array}{cccccccccc} 1 & 1 & 1 & 1 & 1 & & & & & \\ 1 & & 1 & 1 & 1 & & & 1 & & \\ 1 & & 1 & 1 & 1 & & & & & \\ & & & & & 1 & 1 & & & \\ & & & & & 1 & 1 & & & \\ & & & & & & & 1 & 1 & 1 \\ & & & & & & & 1 & 1 & 1 \end{array}\right] \end{array}$$

## 5. Result and discussion

This work considers both an alternative machine and operation types, which sometimes can provide more than one result with the same group technology efficiency. If an alternative machine and operation types are considered in GT, it is more complicated, and the manager is faced with a greater number of selections in deciding which part should be operated on by which machine and which operation type. For instance, there are sixty-four resolutions in example 1, and although $p_1$ could be operated on by $m_1$ or $m_3$ with the same operation type, the method proposed in this article selected $m_1$, because $p_1$ does not need to move between two groups, thus yielding a higher GT efficiency. In addition, one resolution is to let $p_3$ be operated on by $m_1$ with operation type 5 ($o_5$). Another resolution is to let $p_3$ be operated on by $m_1$ with operation types 5 and 6 ($o_5$,$o_6$). The manager has the possibility of selecting which resolution is suitable for the production line, while still benefiting from the GT. In example 2, $p_8$ yields the same GT efficiency whether it is assigned to part family $\{p_4,p_6\}$ or part family $\{p_5,p_9\}$. A manager in a factory could select either family based on the machine load or the real situation.

## 6. Conclusion

CM or GT has many advantages. A major advantage is that the distance or time required for transferring parts is reduced. Many algorithms, such as heuristic, genetic, and neural network algorithms, have been derived in previous studies to realize this reduction. All these traditional GT algorithms focus on "1" or "0," which means that when a machine can operate on a part, then the value is 1; otherwise, the value is 0. Several different machines have the capability to perform the same type of operation. Many machines are not limited to perform only one type of operation. Thus, a suitable machine can be selected to perform more than one type of operation on a part based on the requirement. In this paper, the method of using alternative machines and the method of using a single machine to perform multiple functions are considered. Thus, machine–operation type and operation type- part matrices are introduced. The main step of the algorithm is to form machine cell groups by using a series of logical operations and concepts, and then part families are formed. Two numerical examples are presented to demonstrate the proposed algorithm. It can help the manager to decide which part should be operated on by which machine and which operation type. The results may reveal more than one answer. Thus, two GT efficiency measurements are conducted for testing the proposed algorithm to enable a manager to select the best layout or machines in a factory.

Usually, GT considers only machines and parts, but workers should also be considered. Thus, future work could examine a multi-dimensional GT efficiency that incorporates machines, parts, operation types, and workers.

## References


Agustin-Blas, L. E., Salcedo-Sanz, S., Ortiz-Garcia, E. G., Portilla-Figueras, A., Perez-Bellido, A. M., & Jimenez-Fernandez, S. (2011). Team formation based on group technology: A hybrid grouping genetic algorithm approach. *Computers & Operations Research, 38*(2), 484–495. https://doi.org/10.1016/j.cor.2010.07.006

Alhourani, F. (2016). Cellular manufacturing system design considering machines reliability and parts alternative process routings. *International Journal of Production Research, 54*(3), 846–863. https://doi.org/10.1080/00207543.2015.1083626

Boe, W. J., & Cheng, C. H. (1991). A close neighbor algorithm for designing a cellular manufacturing system. *International Journal of Production Research, 29*(10), 2097–2116. https://doi.org/10.1080/00207549108948069

Bootaki, B., Mahdavi, I., & Paydar, M. M. (2016). New criteria for configuration of cellular manufacturing considering product mix variation. *Computers & Industrial Engineering, 98*, 413–426. https://doi.org/10.1016/j.cie.2016.06.021

Boulif, M., & Atif, K. (2006). A new branch-&-bound-enhanced genetic algorithm for the manufacturing cell formation problem. *Computers & Operations Research, 33*, 2219–2245. https://doi.org/10.1016/j.cor.2005.02.005

Chandrasekharan, M. P., & Rajagopalan, R. (1986). An ideal seed non-hierarchical clustering algorithm for cellular manufacturing. *International Journal of Production Research, 24*(2), 451–464. https://doi.org/10.1080/00207548608919741

Danilovic, M., & Ilic, O. (2019). A novel hybrid algorithm for manufacturing cell formation problem. *Expert Systems with Applications, 135*, 327–350. https://doi.org/10.1016/j.eswa.2019.06.019

Eguia, I., Molina, J. C., Lozano, S., & Racero, J. (2017). Cell design and multi-period machine loading in cellular reconfigurable manufacturing systems with alternative routing. *International Journal of Production Research, 55*(10), 2775–2790. https://doi.org/10.1080/00207543.2016.1193673

Feng, H., Da, W., Xi, L., Pan, E., & Xia, T. (2017). Solving the integrated cell formation and worker assignment problem using particle swarm optimization and linear programming. *Computers & Industrial Engineering, 110*, 126–137. https://doi.org/10.1016/j.cie.2017.05.038

Feng, Y. L., Li, G., & Sethi, S. P. (2018). A three-layer chromosome genetic algorithm for multi-cell scheduling with flexible routes and machine sharing. *International Journal of Production Economics, 196*, 269–283. https://doi.org/10.1016/j.ijpe.2017.12.003

Forghani, K., & Fatemi Ghomi, S. M. T. (2020). Joint cell formation, cell scheduling, and group layout problem in virtual and classical cellular manufacturing systems. *Applied Soft Computing Journal, 97*, 106719. https://doi.org/10.1016/j.asoc.2020.106719

Gupta, N. S., Devika, D., Valarmathi, B., Sowmiya, N., & Shinde, A. (2014). CARI – a heuristic approach to machine-part cell formation using correlation analysis and relevance index. *International Journal of Production Research, 52*(24), 7297–7309. https://doi.org/10.1080/00207543.2014.922709

Hazarika, M., & Laha, D. (2015). Machine-part cell formation for maximum grouping efficacy based on genetic algorithm. *IEEE Workshop on Computational Intelligence: Theories, Applications and Future Directions (WCI)*. https://doi.org/10.1109/WCI.2015.7495521

Hazarika, M., & Laha, D. (2018). Genetic Algorithm approach for Machine Cell Formation with Alternative Routings. *Materials Today Proceedings, 5*, 1766–1775. https://doi.org/10.1016/j.matpr.2017.11.274

Huang, S., & Yan, Y. (2019). Part family grouping method for reconfigurable manufacturing system considering process time and capacity demand. *Flexible Services and Manufacturing Journal, 31*, 424–445. https://doi.org/10.1007/s10696-018-9322-1

Hyer, N., & Wemmerlov, U. (2002). Reorganizing the Factory. *Productivity Press (Chapter, 3)*.

Imran, M., Kang, C., Lee, Y. H., Jahanzaib, M., & Aziz, H. (2017). Cell formation in a cellular manufacturing system using simulation integrated hybrid genetic algorithm. *Computers & Industrial Engineering, 105*, 123–135. https://doi.org/10.1016/j.cie.2016.12.028

Kumar, R., & Singh, S. P. (2017). A similarity score-based two-phase heuristic approach to solve the dynamic cellular facility layout for manufacturing systems. *Engineering Optimization, 49*(11), 1848–1867. https://doi.org/10.1080/0305215X.2016.1274205

Kusiak, A., & Cho, M. (1992). Similarity coefficient algorithms for solving the group technology problem. *International Journal of Production Research, 30*(11), 2633–2646. https://doi.org/10.1080/00207549208948181

Laha, D., & Hazarika, M. (2017). A heuristic approach based on Euclidean distance matrix for the machine-part cell formation problem. *Materials Today Proceedings, 4*, 1442–1451. https://doi.org/10.1016/j.matpr.2017.01.166

Li, M. L. (2007). Efficiency measurement for multi-dimensional group technology. *The International Journal of Advanced Manufacturing Technology, 35*, 621–632. https://doi.org/10.1007/s00170-006-0734-9

Li, M. L. (2020). Arrangement of Machines Under Spatial Constraint by Using a Novel Algorithm. *IEEE Access, 8*, 144565–144574. https://doi.org/10.1109/ACCESS.2020.3013907

Li, M. L., & Parkin, R. E. (2002). Expansion of group technology to handle logical operations. *International Journal of Production Research, 40*(16), 4169–4194. https://doi.org/10.1080/00207540210148899

Lian, J., Liu, C. G., Li, W. J., Evans, S., & Yin, Y. (2014). Formation of independent manufacturing cells with the consideration of multiple identical machines. *International Journal of Production Research, 52*(5), 1363–1400. https://doi.org/10.1080/00207543.2013.843797

Liang, X. X., Liu, M., Feng, Y. B., Wang, J. B., & Wen, L. S. (2019). Solution algorithms for single-machine resource allocation scheduling with deteriorating jobs and group technology, Engineering Optimization, online July 19, 2019. https://doi.org/10.1080/0305215X.2019.1638920.

Liu, C., Wang, J., & Leung, J. Y. T. (2018). Integrated bacteria foraging algorithm for cellular manufacturing in supply chain considering facility transfer and production planning. *Applied Soft Computing, 62*, 602–618. https://doi.org/10.1016/j.asoc.2017.10.034

Lu, Y. Y., Wang, J. B., Ji, P., & He, H. (2017). A note on resource allocation scheduling with group technology and learning effects on a single machine. *Engineering Optimization, 49*(9), 1621–1632. https://doi.org/10.1080/0305215X.2016.1265305

McAuley, J. (1972). Machine grouping for efficient production. *The Production Engineer*, 53–57.

Mehdizadeh, E., Niaki, S. V. D., & Rahimi, V. (2016). A vibration damping optimization algorithm for solving a new multi-objective dynamic cell formation problem with workers training. *Computers & Industrial Engineering, 101*, 35–52. https://doi.org/10.1016/j.cie.2016.08.012

Miao, C. (2019). Parallel-Batch Scheduling with Deterioration and Group Technology. *IEEE Access, 7*, 119082–119086. https://doi.org/10.1109/ACCESS.2019.2936004

Mohammadi, M., & Forghani, K. (2016). Designing cellular manufacturing systems considering S-shaped layout. *Computers & Industrial Engineering, 98*, 221–236. https://doi.org/10.1016/j.cie.2016.05.041

Nalluri, M. S. R., Kannan, K., & Gao, X. Z. (2019). An efficient hybrid meta-heuristic approach for cell formation problem. *Soft Computing, 23*(19), 9189–9213. https://doi.org/10.1007/s00500-019-03798-7

Navaei, J., & ElMaraghy, H. (2014). Grouping Product Variants Based on Alternate Machines for Each Operation. *Procedia CIRP, 17*, 61–66. https://doi.org/10.1016/j.procir.2014.01.124

Parkin, R. E., & Li, M. L. (1997). The Multi-Dimensional Aspects of a Group Technology Algorithm. *International Journal of Production Research, 35*(7), 2345–2358. https://doi.org/10.1080/002075497194886

Qin, H., Zhang, Z., & Bai, D. (2016). Permutation flowshop group scheduling with position-based learning effect. *Computers & Industrial Engineering, 92*, 1–15. https://doi.org/10.1016/j.cie.2015.12.001

Rahimi, V., Arkat, J., & Farughi, H. (2020). A vibration damping optimization algorithm for the integrated problem of cell formation, cellular scheduling, and intercellular layout. *Computers & Industrial Engineering, 143*, 106439. https://doi.org/10.1016/j.cie.2020.106439

Shahdi-Pashaki, S., Teymourian, E., Kayvanfar, V., Komaki, G. M., & Sajadi, A. (2015). Group technology-based model and cuckoo optimization algorithm for resource allocation in cloud computing. *IFAC, 48–3*, 1140–1145. https://doi.org/10.1016/j.ifacol.2015.06.237

Singh, N., & Rajamani, D. (1996). Cellular Manufacturing Systems: Design, Planning and Control (1st ed.). Chapman & Hall (Chapter 1).

Suzic, N. (2012). Customizing Products through Application of Group Technology: A Case Study of Furniture Manufacturing. *Journal of Mechanical Engineering, 58*(12), 724–731.

Wemmerlov, U., & Hyer, N. L. (1989). Cellular manufacturing in the U.S. industry: A survey of users. *International Journal of Production Research, 27*(9), 1511–1530. https://doi.org/10.1080/00207548908942637

Won, Y. (2020). On Solving Large-Size Generalized Cell Formation Problems via a Hard Computing Approach Using the PMP. *Applied sciences, 10*, 3478. https://doi.org/10.3390/app10103478

Won, Y., & Logendran, R. (2015). Effective two-phase p-median approach for the balanced cell formation in the design of cellular manufacturing system. *International Journal of Production Research, 53*(9), 2730–2750. https://doi.org/10.1080/00207543.2014.977457

Zhang, X., Liao, L., Zhang, W., Cheng, T. C. E., Tan, Y., & Ji, M. (2018). Single-machine group scheduling with new models of position-dependent processing times. *Computers & Industrial Engineering, 117*, 1–5. https://doi.org/10.1016/j.cie.2018.01.002

Zohrevand, A. M., Rafiei, H., & Zohrevand, A. H. (2016). Multi-objective dynamic cell formation problem: A stochastic programming approach. *Computers & Industrial Engineering, 98*, 323–332. https://doi.org/10.1016/j.cie.2016.03.026