Petroleum Business Review, Vol. 8, No. 1, pp. 76–97, January 2024

Application of Cell Formation Problem for Optimal Job Grouping in Oil and Gas Megaprojects Using a Harmony Search Algorithm

Mohammad Ali Hatefi[1*] and Fatemeh Davoodi[2]

[1] Petroleum University of Technology, Tehran, Iran
[2] M.S. Graduate, Petroleum University of Technology, Tehran, Iran

Highlights

- Defining job grouping problem as a CFP model.
- Introducing an HSA to solve CFP.
- Demonstrating considerable performance of the proposed HSA.

Received: January 21, 2024; *revised:* February 12, 2024; *accepted:* March 11, 2024

Abstract

Cell formation problem (CFP) is a famous issue in group technology, where an analyst aims at forming groups of machines/parts (so-called cells) in such a way that machines in every cell process as many as possible parts from this cell and as less as possible parts from other cells. In this article, a harmony search algorithm (HSA) is designed to solve the CFP. To evaluate the performance of the proposed algorithm, we examined 35 sample classical problems. The experimental results indicate the desirable performance of the proposed algorithm. In addition, the applicability of the proposed algorithm is demonstrated by a real-world case in project human resources management. Since the megaprojects in the oil industry require a wide range of jobs, grouping individuals with considering their skills is an important challenge. The paper defines the job grouping problem as a CFP and solves a real-world problem using the proposed HSA.

Keywords: CFP, HSA, Job grouping, Oil and gas.

How to cite this article
Hatefi, M.A, Davoodi, F, Application of Cell Formation Problem for Optimal Job Grouping in Oil and Gas Megaprojects, Using a Harmony Search Algorithm, Petroleum Business Review, Vol. 8, No. 1, p. 76–97, 2024. DOI: 10.22050/pbr.2024.436980.1324

1. Introduction

Project managers, after identifying project requirements, need efficient human resources management to reach project goals. Project managers of megaprojects encounter a wide range of jobs and skills, so grouping individuals with concerning skills is inevitable. According to Flyvbjerg (2017), megaprojects are large-scale, complex ventures that typically cost $1 billion or more, take many years to develop and build, involve multiple public and private stakeholders, are transformational, and impact millions of people. Job grouping in megaprojects, especially when individuals are separated from each other in different geographical locations, causes significant advantages such as economic savings, productivity in training, teamwork enhancement, and better communication. On the other words, job grouping helps project managers to have a clear image of similar business groups. Simply, herein, the challenge is that individuals who have similar and related skills are classified in the same group. Of course, this is easily

* Corresponding author:
Email: hatefi@put.ac.ir

possible for small projects with fewer individuals and jobs, but in megaprojects with a significant number of human resources and skills, having a reliable mathematical method for job grouping is a very important challenge. To this end, the idea in this paper is to employ the famous cell formation problem (CFP). The notion is to define a matrix similar to the CFP, in which skills are instead of machines, and individuals are instead of parts. It should be noted that this idea has not been taken yet.

Cell formation problem is a known challenge in group technology. This problem is to form groups of machines/parts (so-called cells) so that machines in every cell process as much as possible parts from this cell and as less as possible parts from other cells (Bychkov et al., 2014). The cells are shaped to get the many potential advantages such as reduced setup times, reduced in-process inventories, enhanced product quality, minimal lead-time, reduced tool requirements, and improved productivity (Rostami et al., 2020). In other words, if the cells can operate independently, the costs of operation, transportation, overtime, and training will decrease. The CFP is an NP-hard optimization problem and cannot be solved using exact methods in a reasonable time. In the relevant literature, several methods have been proposed to solve the CFP. These methods can be broadly categorized into two branches: exact algorithms and approximate algorithms. Exact algorithms are capable of finding the optimal solution accurately; however, they are not efficient enough for big problems, and their execution time increases exponentially according to the dimension of the problem. Approximate algorithms, such as heuristics and metaheuristics, are to find good (near-optimal) solutions in a short time even for difficult problems.

Numerous studies have been published in the field of the CFP. There are few efforts to develop exact methods for the CFP, such as optimal tree search based on graph partitioning concept (Spiliopoulos and Sofianopoulou (1998), branch and bound algorithm (B&B) (Arkat et al., 2012), separation algorithm based on bicluster graph editing problem (BGEP) (Pinheiro et al., 2016), possibilistic mixed-integer programming model (Hashemoghli et al., 2019), and various mathematical programming models (Dehnavi-Arani et al., 2020; Rafee and Mohamaditalab, 2020; Xue and Offodile, 2020; Tavanayi et al., 2021). Given the difficulty in searching optimal solutions for large-sized CFP, several heuristic algorithms have been developed to reach approximate solutions. The following are some examples: water flow-like algorithm (WFA) (Wu et al., 2010), ant colony optimization (ACO) (Li et al., 2010; Rabbani et al., 2017), simulated annealing (SA) (Pailla, et al. 2010; Kamalakannan et al., 2019), genetic algorithm (GA) (Elbenani et al., 2012; Shashikumar et al., 2019; Hazarika, 2023), electromagnetism algorithm (EMA) (Wei et al., 2012), scatter search (SS) (Tavakkoli-Moghaddam et al., 2012), particle swarm optimization (PSO) (Kao and Lin, 2012; Ayough and Khorshidvand, 2019), iterated local search (ILS) (Brusco, 2015), the ILS coupled with a variant of the variable neighborhood descent (VND) (Martins et al., 2015), bacteria foraging algorithm (BFA) (Liu et al., 2015), imperialist competitive algorithm (ICA) (Borghei et al., 2015), league championship algorithm (LCA) (Noktehdan et al., 2016), clonal selection algorithm (CSA) (Karoum and Elbenani, 2017), artificial bee colony (ABC) (Arunagiri et al., 2017), tabu search (TS) (Kamalakannan and Pandian, 2018), cell formation optimization (CFOPT) (Danilovic and Ilic, 2019), cat swarm optimization (CSO) (Soto et al., 2019a), human behavior-based algorithm (HBBA) (Soto et al., 2019b), butterfly and firefly methods (Nagaraj et al., 2020), a metaheuristic algorithm based on the gray wolf optimization (GWO) (Saeidi and Nikakhtar, 2022), black widow optimization (BWO) (Figueroa-Torrez et al., 2023), and some mixed methods such as hybrid GA and ICA algorithms (Bagheri and Bashiri, 2014) and combined PSO and SA algorithms (Behnia et al., 2019). Finally, it should be noted that to the best of the author's knowledge which is based on reviewing the relevant literature, there is no well-established and journal-published work employing the harmony search algorithm (HSA) to solve the CFP. Therefore, one of the objectives of this study is to fill this gap.

This paper proposes a new HSA to solve the CFP. The remainder of the paper is organized as follows: Section 2 offers some preliminaries around the CFP, and section 3 provides the research materials and methods, including a complete description of the proposed HSA and a solution coding for the HSA in the CFP. Experimental and computational analysis of the proposed algorithm is given in section 4. Section 5 deals with a real-world application of CFP problem for job grouping in megaprojects, and draws up the results. Section 6 concludes the paper.

2. Preliminaries

The CFP matrix is comprised of numbers 0 and 1. Its rows and columns indicate p parts ($i = 1, \dots, p$) and m machines ($j = 1, \dots, m$), respectively. The ij component is equal to 1, if part i has an operation on machine j; otherwise, it equals 0. For example, suppose we have a CFP with 5 parts (p = 5) and 6 machines (m = 6) as the left matrix in Figure 1. The goal is to find a matrix with the same number of rows and columns, and the same number of 0s and 1s so that the 1s have been concentrated in one cell. In this example, we reach the answer with a given heuristic method as the right matrix in Figure 1. This matrix, using two lines, is divided into four blocks: two diagonal blocks and two off-diagonal blocks. The diagonal blocks indicate the cells. Let's depict the number of cells by n, so in this example n equals 2. As shown in the matrix, parts 1, 4, and 3 with corresponding machines 1, 6, and 3 are placed in the first cell, and parts 2 and 5 with corresponding machines 2, 5, and 4 constitute the second cell. We see that a 1 indicates part 1, and machine 4 does not belong to any cell; therefore, machine 4 can work overtime in the second cell and can compensate for what part 5 cannot do in that cell. The 1's that are placed outside of all the cells are called exceptions, and the 0's placed inside of a cell are called void.

	M_1	M_2	M_3	M_4	M_5	M_6
P_1	1	0	0	1	0	1
P_2	0	1	1	0	0	1
P_3	1	0	1	0	0	1
P_4	1	0	0	1	1	0
P_5	0	1	0	0	1	0

	M_1	M_6	M_3	M_2	M_5	M_4
P_1	1	1	0	0	0	1
P_4	1	1	1	0	0	0
P_3	1	1	1	0	0	0
P_2	0	0	0	1	1	1
P_5	0	0	0	1	1	0

Figure 1

A sample of the CFP matrix (left: ungrouped, right: grouped)

There are several criteria to assess the quality of a given solution. The original criterion is termed grouping efficiency as Equation (1), which was suggested by Chandrasekharan and Rajagopalan (1986).

$$\eta = q \times g_1 + (1 - q) \times g_0 \tag{1}$$

where q is a weighting factor, g_1 is the ratio of the number of 1's in the diagonal blocks to the total number of elements in the diagonal blocks of the matrix, and g_0 is the number of 0's in the off-diagonal blocks to the total number of elements in the off-diagonal blocks of the final matrix. For instance, for the right matrix in Figure 1, g_1 = 13/15 and g_2 = 14/15. This criterion functions weakly for large-size matrices. A different criterion was proposed by Kumar and Chandrasekharan (1990), namely grouping efficacy as Equation (2), which overcomes the grouping efficiency weaknesses.

$$\zeta = \frac{e - e_0}{e + e_v} \tag{2}$$

where e is the total number of 1's in the CFP matrix, e_v is the number of voids, and e_0 is the number of exceptions. For instance, for the right matrix in Figure 1, ζ is equal to (14 – 1)/(14 + 1).

3. Methodology

3.1. Harmony search algorithm

The HSA was proposed by Geem et al. (2001). This algorithm is inspired by the explicit principles of harmony improvisation. When musicians compose a harmony, they usually test various possible combinations of the music pitches stored in their memory, which can be considered an optimization process of adjusting the input (pitches) to obtain the optimal output (perfect harmony). In the HSA, each solution is called a harmony and is presented by a vector. A primary population of harmony memory size (HMS) of a number of harmony vectors is randomly generated and stored in harmony memory (HM), depending on the objective function values. Then, a new harmony is created. If the new harmony is better than the worst existing harmony in the HM, it is replaced by the new harmony, and then a resort of the harmonies is done. This process continues until reaching a stopping criterion. The HSA consists of six steps as follows:

Step 1: Define the objective function. For the proposed algorithm in this study, we select the grouping efficacy measure as Equation (2) to be maximized.

Step 2: Tune (adjust) the parameters: Harmony memory size, harmony memory considering rate (HMCR), pitch adjusting rate (PAR), band width (BW), and the maximum of the iteration number (MaxIt);

Step 3: Initialize the HM, i.e., the random generation of the harmonies;

Repeat steps 4–6 until a termination criterion is met.

Step 4: Improvise (or generate) a new harmony;

Step 5: Compute the new harmony objective function;

Step 6: Update the HM.

In order to manage utilization from the HM, a parameter called HMCR ∈ [0,1] is defined. We define the probability of random improvising by Equation (3). The algorithm at $\mathrm{P_{random}}\%$ of the time randomly improvises new harmony, and at $1 - \mathrm{P_{random}}\%$ of the time generates new harmony.

$$P_{random} = 1 - HMCR \tag{3}$$

If HMCR is too small (close to 0), instead of benefiting from the HM, random harmonies are used; thus, not only may the convergence to the final solution be very slow, but also the solution variety may be increased. Conversely, if this rate is too large (close to 1), existing harmonies in the HM can then be used to generate new harmony. In this case, other harmonies in the HM may not be well scanned, so this may lead to algorithm trapping in local optimal. Typically an HMCR between 0.70 to 0.95 is used.

If the algorithm decides to generate new harmony based on existent harmonies in the HM, other parameters called PAR and BW (values from the interval (0 1)) play their roles. A small PAR with a narrow bandwidth can cause a slowdown in harmony search convergence. However, maintaining a smaller PAR and larger BW causes a diversity of harmony, which can quickly find local optimal solutions. Anyway, pitch-adjusted frequency corresponds to the production of a roughly different solution in the HSA. Theoretically, the pitch can be either in a linear or nonlinear form. Practically, linear adjusting is used, so we have Equation (4).

$$x_{new} = x_{old} + BW \times \varepsilon \quad (4)$$

where x_{old} is an existing solution from the HM, and x_{new} is the newly generated harmony. Pitch adjusted basically generates a new solution around the existing solution by a small pitch change with a small random value. Here, ε is a random value from the interval [–1, 1]. Taking PAR into account, we can control the variation degree. A small PAR can cause a slowdown in the HSA convergence. On the other hand, considering a high value for this parameter may also lead to a great dispersion of solutions around optimal points, like a random search. At most times, the typical value for PAR is a number between 0.1 and 0.5. The real probability of applying pitch adjusted is equal to HMCR × PAR. The pseudo-code of the HSA is presented in Figure 2.

- **Begin**
- Define objective function
- Tune HMCR, PAR, BW, and other parameters
- Improvise random harmonies, as many as MaxIt (maximum of the iteration number)
- Using improvised harmonies, constitute the HM (a sorted array of harmonies)
- Set *u* = 0
- **While** *u* < MaxIt
- Set *u* = *u* +1
- Set *v* = 0
- **While** *v* < HMS
- Set *v* = *v* +1
- Generate a random number between 0 and 1 and name it *rand*
- **If** *rand* < HMCR, then choose harmony *v* from the HM
- Generate a random number between 0 and 1 and name it *rand*
- **If** *rand* < PAR then
- generate a new harmony using pitch adjusted rule and go to comparing point
- **Else** nothing done
- **End if**
- **Else** improvise a random harmony
- **End if**
- **End while**
- Comparing point:
- If the worst solution in the HM is worse than the new harmony, then
- replace it, and, after that, resort to the harmonies in the HM
- **End while**
- Take the best solution from the HM
- **End**

Figure 2

The pseudo-code of the HSA

3.2. CFP solution coding in HSA

At first, the number of cells (to be constructed) is defined according to Equation (6).

$$n = \lceil r \times (n_{max} - 1) \rceil + 1 \quad (6)$$

where n_{max} is the maximum number of cells, *r* denotes a random number ranging from 0 to 1, and $\lceil \ \rceil$ depicts the ceiling function. This statement ensures that the number of constructed cells is always between 1 and n_{max}. A solution array in the HSA is shown by a chromosome string including $p + m +$

1 genes, p genes indicating parts, m genes indicating machines, and a gene indicating the number of cells. Each gene (i.e., the entry of the array), referring to as a part or machine, is assigned a random number between 0 and 1; we call them random values. The number of cells (n) is inserted at the last right gene of the chromosome. For example, Figure 4 depicts a randomly generated chromosome with p = 5, m = 6, and n = 2. Having a given array, the cell number of a part or machine is determined by Equation (7).

$$Cell\ no = \lceil random\ value \times n \rceil \tag{7}$$

In this statement, the random value represents the numbers in each entry of the produced array. For example, in Figure 3, this value is equal to 0.54 for P1, so the cell number of P1 is $\lceil 0.54 \times 2 \rceil$ or 2. This equation ensures that the made assignment for each part or machine to a cell is placed in one of the predicted cells that are proportional to the initial random value generated by it. Therefore, the output of this example is shown in Figure 4.

P1	P2	P3	P4	P5	M1	M2	M3	M4	M5	M6	*n*
0.54	0.63	0.66	0.92	0.27	0.76	0.09	0.29	0.62	0.34	0.57	2

Figure 3

A sample solution array in the HSA

P1	P2	P3	P4	P5	M1	M2	M3	M4	M5	M6	*n*
2	2	2	2	1	2	1	1	2	1	2	2

Figure 4

The array in which each gene is assigned to a cell

4. Experimental analysis

In this section, the parameter tuning procedure of the HSA is firstly described. Then, the obtained results of the HSA are compared with the presented results by the other solving methods in the related research background.

4.1. Tuning parameters

Taguchi is a well-founded method to tune the algorithm parameters using T-test problems ($k = 1, \dots, T$). It seeks to determine the best levels of parameters. For the sake of double-checking the results, we use two measures here. Firstly, the obtained results from each test problem are transformed to a rate called signal-to-noise (S/N) as Equation (8). At this rate, the desirable value (average) and the undesirable value (standard deviation) are called the signal and the noise, respectively. The target is to find the parameter levels that maximize the S/N rate.

$$S/N\ ratio = -10log\frac{1}{T}\left[\sum_{k=1}^{T} objective\ function_k^2\right] \tag{8}$$

Secondly, for each test problem, the value of the objective function is transformed to relative percentage deviation (RPD) as Equation (9).

$$RPD = \frac{Benchmark_{solution} - Algorithm_{solution}}{Benchmark_{solution}} \tag{9}$$

where $Algorithm_{solution}$ is the obtained objective function value for each test problem, and $Best_{solution}$ is the last best value reported in the relevant literature. Lower values for RPD indicate the better quality of the response.

A considerable attribute of the Taguchi method is the fact that it does not examine all the possible test problems, i.e. it is not a full-factorial method. As a matter of fact, in the Taguchi method, the best level of each parameter can be achieved only by examining a fraction of test problems, so the Taguchi is a fractional-factorial method. In the proposed HSA, four parameters are tuned. These parameters are MaxIt, HMS, HMCR, and PAR. The levels of these parameters for the HSA are given in Table 1.

Table 1

The level of the parameters in the HAS

Factor	Level 1	Level 2	Level 3
MaxIt	1000	3000	5000
HMS	30	60	90
HMCR	0.3	0.6	0.9
PAR	0.3	0.6	0.9

In Table 1, each factor has three levels. According to a full-factorial design, 81 (=3 × 3 × 3 × 3) different test problems should be defined and solved. However, we will see that based on the Taguchi method, only 9 test problems (out of 81) are taken into account. Given the levels of factors (as in Table 1), the orthogonal array of the Taguchi method is L9. To perform calculations, we used the MINITAB-16 software. Table 2 gives the final results.

Table 2

Test information and S/N rates

Test problem no.	MaxIt	HMS	HMCR	PAR	RPD	S/N rate
1	1	1	1	1	0.32	9.87
2	1	2	2	2	0.19	14.37
3	1	3	3	3	0.05	25.84
4	2	1	2	3	0.10	20.39
5	2	2	3	1	0.05	26.85
6	2	3	1	2	0.28	10.93
7	3	1	3	2	0.01	37.25
8	3	2	1	3	0.25	12.14
9	3	3	2	1	0.10	19.65

The S/N averages for each level of the parameters are presented in Figure 5 and Table 3. For example, in the first level of the HMS diagram, the S/N average is calculated as (9.87 + 20.39 + 37.25)/3 = 22.51. Based on this diagram, the best-obtained levels for MaxIt, HMS, HMCR, and PAR are 3, 1, 3, and 2, respectively. This result is confirmed by both the RPD data and S/N rates displayed in Table 2.

Moreover, based on the obtained responses for S/N rates, two additional results are achieved: Delta and Rank (see Table 3). Delta is the difference between the highest and lowest average response values for each parameter. Ranks are assigned based on Delta values: Rank 1 to the highest Delta value, Rank 2 to the second highest, and so on. Consequently, the most effective parameters are MaxIt, HMS, HMCR, and eventually PAR, respectively. This result can be also evaluated concerning the gradients of the

diagrams presented in Figure 5. Given the results, the parameters are adjusted at MaxIt = 5000, HMS = 30, HMCR = 0.9, and PAR = 0.6.

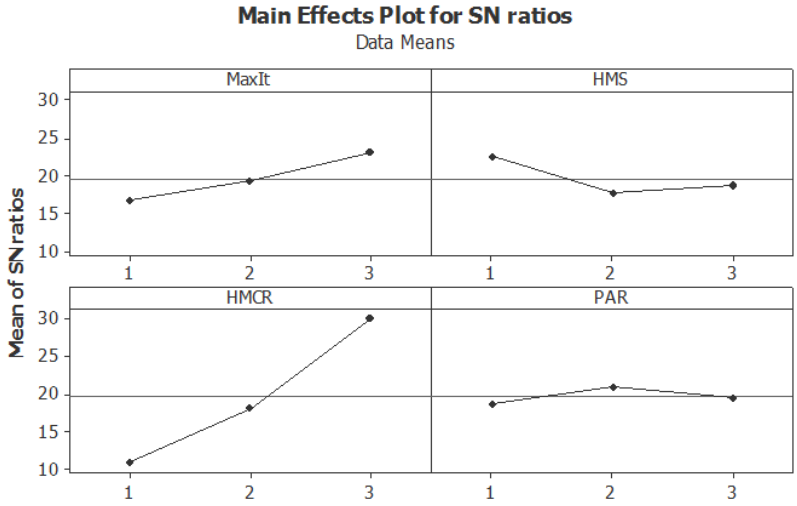


Figure 5

The average S/N rates for four parameters

Table 3

The Delta and Ranks results in the Taguchi test

Level	MaxIt	HMS	HMCR	PAR
1	16.69	22.51	10.98	18.79
2	19.39	17.79	18.14	20.85
3	23.01	18.8	29.98	19.46
Delta	6.32	4.72	19	2.06
Rank	2	3	1	4

4.2. Experimental results

To test the performance of the proposed HSA, 35 instance problems (i.e., the classical CFP dataset) available in Goncalves and Resende (2004) are employed, since they have been used as a benchmark in many researches. These problems have been used for several years by CFP investigators for computational experiments and response comparison. The summary data on the instance problems are presented in Table 4.

Table 4

Instance problems information

#	Source	Matrix size
1	King and Nakornchai (1982)	5 × 7
2	Waghodekar and Sahu (1984)	5 × 7
3	Seifoddini (1989)	5 × 18
4	Kusiak and Cho (1992)	6 × 8

#	Source	Matrix size
5	Kusiak and Chow (1987)	7 × 11
6	Boctor (1991)	7 × 11
7	Seifoddini and Wolfe (1986)	8 × 12
8	Chandrasekharan and Rajagopalan (1986a)	8 × 20
9	Chandrasekharan and Rajagopalan (1986b)	8 × 20
10	Mosier and Taube (1985a)	10 × 10
11	Chan and Milner (1982)	10 × 15
12	Askin and Subramanian (1987)	14 × 24
13	Stanfel (1985)	14 × 24
14	McCormick et al. (1972)	16 × 24
15	Srinivasan et al. (1990)	16 × 30
16	King (1980)	16 × 43
17	Carrie (1973)	18 × 24
18	Mosier and Taube (1985b)	20 × 20
19	Kumar et al. (1986)	20 × 23
20	Carrie (1973)	20 × 35
21	Boe and Cheng (1991)	20 × 35
22	Chandrasekharan and Rajagopalan (1989)	24 × 40
23	Chandrasekharan and Rajagopalan (1989)	24 × 40
24	Chandrasekharan and Rajagopalan (1989)	24 × 40
25	Chandrasekharan and Rajagopalan (1989)	24 × 40
26	Chandrasekharan and Rajagopalan (1989)	24 × 40
27	Chandrasekharan and Rajagopalan (1989)	24 × 40
28	McCormick et al. (1972)	27 × 27
29	Carrie (1973)	28 × 46
30	Kumar and Vannelli (1987)	30 × 41
31	Stanfel (1985)	30 × 50
32	Stanfel (1985)	30 × 50
33	King and Nakornchai (1982)	36 × 90
34	McCormick et al. (1972)	37 × 53
35	Chandrasekharan and Rajagopalan (1987)	40 × 100

The proposed algorithm was coded in MATLAB software and was run on an Intel processor having three cores, CPU M370 2.40GHz, and 4 GB RAM. To compare the obtained results with the provided results in other works, the RPD criterion is used. Table 5 lists the experimental results of the HSA (with

no cell size constraints) and the last best-known solution reported in the literature, taken from Danilovic and Ilic (2019), who claim their algorithm, the CFOPT, is the best algorithm related to the known algorithms for the CFP.

Table 5

The results of HSA via the best known-solution in the literature

#	The best known-solution	HSA	RPD (%)
1	82.35	82.35	0.00%
2	69.57	69.57	0.00%
3	80.85	80.85	0.00%
4	79.17	79.17	0.00%
5	60.87	59.26	2.64%
6	70.83	70.83	0.00%
7	69.44	69.44	0.00%
8	85.25	85.25	0.00%
9	58.72	56.88	3.13%
10	75	71.88	4.16%
11	92	92.00	0.00%
12	74.24	72.06	2.94%
13	72.86	71.83	1.41%
14	53.33	52.75	1.09%
15	69.92	68.99	1.33%
16	58.04	57.53	0.88%
17	57.73	57.43	0.52%
18	43.97	43.18	1.80%
19	50.81	50.81	0.00%
20	79.38	78.11	1.60%
21	58.79	57.98	1.38%
22	100	100.00	0.00%
23	85.11	85.11	0.00%
24	73.51	73.51	0.00%
25	53.29	53.27	0.04%
26	48.95	48.66	0.59%
27	47.26	46.90	0.76%
28	54.82	53.84	1.79%
29	47.85	46.91	1.96%
30	63.31	62.74	0.90%
31	60.12	59.77	0.58%
32	50.84	50.83	0.02%
33	48.29	44.61	7.62%
34	61.36	59.96	2.28%
35	84.03	84.03	0.00%
	Average		1.13%

According to the results presented in Table 5 and values presented in column RPD, it is observed that RPD has insignificant values. On average, RPD is roughly equal to 1.13%, indicating an acceptable closeness to the best-known solutions; in the worst case, it is equal to 7.62%.

Table 6 presents the optimal objective function values of instance problems obtained by an exact algorithm called the iterative method (Pinheiro et al., 2016). Instance problems with NA (in the iterative method column) indicate that there are no data about optimal solutions by this method, i.e., we could not get an optimal solution for those problems. The average RPD is 0.84% which shows a highly desirable performance of the proposed HSA.

Table 6

The results by the HSA in comparison with the optimal results

#	Iterative method	HSA	RPD (%)
1	82.35	82.35	0.00%
2	69.57	69.57	0.00%
3	80.85	80.85	0.00%
4	79.17	79.17	0.00%
5	60.87	59.26	2.64%
6	70.83	70.83	0.00%
7	69.44	69.44	0.00%
8	85.25	85.25	0.00%
9	58.72	56.88	3.13%
10	75.00	71.88	4.16%
11	92.00	92.00	0.00%
12	74.24	72.06	2.94%
13	72.86	71.83	1.41%
14	53.33	52.75	1.09%
15	69.92	68.99	1.33%
16	58.04	57.53	0.88%
17	57.73	57.43	0.52%
18	NA	43.18	-
19	50.81	50.81	0.00%
20	79.38	78.11	1.60%
21	58.79	57.98	1.38%
22	100.00	100	0.00%
23	85.11	85.11	0.00%
24	73.51	73.51	0.00%
25	53.29	53.27	0.04%
26	NA	48.66	-
27	NA	46.90	-
28	NA	53.84	-
29	NA	46.91	-
30	63.31	62.74	0.90%
31	60.12	59.77	0.58%
32	NA	50.83	-

#	Iterative method	HSA	RPD (%)
33	NA	44.61	-
34	NA	59.96	-
35	84.03	84.03	0.00%
	Average		0.84%

Table 7 represents the CPU time (runtime) for the HSA and iterated methods. Given the runtime, the HSA performance is remarkable in terms of the short period. The runtime of the proposed algorithm for various problems is about a few seconds which indicates a good performance of the HSA to solve the problems with different sizes.

Table 7

The CPU time comparison (in seconds)

#	HSA	Iterative method
1	5.46	0.16
2	5.05	0.07
3	7.32	0.09
4	5.37	0.02
5	6.34	0.14
6	6.43	0.29
7	6.43	0.18
8	8.07	2.06
9	8.24	81.46
10	6.59	0.03
11	7.65	0.01
12	10.31	0.49
13	9.96	0.49
14	10.32	600.98
15	12.36	7.24
16	15.59	1156.23
17	11.86	87.13
18	11.16	-
19	11.66	23928.7
20	15.11	2145.24
21	14.86	1.78
22	17.03	0.02
23	16.50	10.08
24	16.65	17.46
25	16.63	371233
26	16.20	-
27	17.01	-
28	13.37	-

#	HSA	Iterative method
29	17.73	-
30	18.05	183.71
31	20.23	13807.5
32	19.91	-
33	31.98	-
34	22.87	-
35	35.13	325.53

5. A case study of job grouping

One of the fundamental issues in the oil and gas industries is the need for exploration and production (E&P) companies to perform megaprojects which have various aspects from the technical, financial, human resources, and economic viewpoints. Hence, these megaprojects usually need a wide range of skills and capabilities.

The selected real-world case study for grouping its jobs is a drilling megaproject defined in an Iranian E&P company. Drilling is a complex operation that consists of boring a wellbore to extract oil or gas. This operation includes various stages such as boring, circulation, and casing. For this process, the drilling rig, a complex integrated system containing the equipment such as mud tanks and pumps, derrick, drill string, and power generation equipment, is installed (Hatefi and Balilehvand, 2023).

The case study of the megaproject includes 47 individuals (i.e., manpower) and 71 jobs such as administrative manager, security manager, financial assessor, auditor, geologist, oil engineer, earth surveyor, cargo operator, drilling operator, mining machine operator, chemical systems operator, crane operator, rig operator, and crane driver. Table 8 presents the CFP matrix for this problem. The HSA was carried out in two different modes: (I) without limitation for the number of cells and (II) with a limitation for the number of cells, i.e., with $n_{max} = 4$. In mode (I), the optimal outcome of the algorithm is given in Table 9, in which the optimal value is 0.5912. Further, in mode (II), the optimal matrix with an optimal value of 0.5982 is obtained as listed in Table 10.

Table 8

The CFP matrix for the case study

		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
1	senioer manager of the organization or institution		1																														1																
2	Administration Manager				1																																											
3	Sales Manager					1																																										
4	financial manager						1																																										
5	quality control manager							1	1	1																																								
6	Director of Purchasing and Procurement										1		1																																				
7	Director of Transportation (Transportation)													1												1																								
8	Warehousing and Distribution Manager														1																																		
9	Physical Distribution Manager (Logistics)														1	1																																
10	Payroll and Benefits Manager																1	1																															
11	Human Resources Manager																1	1																															
12	Director of Education and Human Resources Development																1	1																															
13	Manager of compliance with regulations and standards						1	1	1																																							
14	Security manager																		1																													
15	Capital Loss Prevention Manager						1	1																																								
16	Responsible for the purchase of goods and equipment									1	1	1																																			
17	Responsible for the purchase									1	1	1																																			
18	Human Resources Specialist																	1																													
19	Logistics and supply chain specialist														1	1				1																												
20	Logistics and Supply Chain Engineer														1	1				1																												
21	Logistics and Supply Chain Analyst														1	1				1																												
22	Management Analyst and Consultant		1	1																																											
23	Accountant						1														1																										
24	auditor						1														1																										
25	Financial Assessor						1														1		1																								
26	Personal and corporate tax advisor																				1		1																								
27	Earth Surveyor																							1	1																						
28	Transport (Traffic) Engineer													1												1																						
29	Mining and Geological Engineer																										1	1	1																		
30	Oil Engineer																													1	1	1															
31	Survey technician																							1	1																						
32	geologist																										1	1	1																		
33	Geophysical technician																										1	1	1																		
34	Technician and technical expert of geological experiments																										1	1	1																		
35	Expert of preparing the declaration		1		1																												1														
36	Accounting Officer, Documentary																				1																										
37	Accounting and auditing staff																				1																										
38	Procurement Officer										1	1	1																																			
39	Cargo and freight forwarding operator													1																																		
40	Oil and gas drilling rig operator																													1	1	1															
41	Oil and gas drilling rig operator																													1	1																
42	Oil and gas and mining equipment operator																													1	1	1															
43	Mining machine operator																													1	1	1															
44	Mining fire chief																										1	1		1	1	1		1												
45	User of continuous drilling machines																										1	1	1																	
46	Mining and gap mine machinery operator																										1	1	1																	
47	Mining Crusher																										1	1	1																	
48	mine worker																										1	1	1																	
49	Simple petrochemical worker																																			1	1									
50	simple mining worker																										1	1	1									1			1			1			
51	rig Maker																										1	1	1										1							
52	Chemical systems operator																																			1	1		1	1						
53	Gas facilities operator																																			1	1		1							
54	Operator of fuel pump and refinery systems																																			1	1		1							
55	Bio-fuel processing technician																																							1	1					
56	Technician and technical expert of the methane gas collection system of landfills																																							1	1					
57	Technician and technical expert of bio-mass power plant																																							1	1					
58	Technician and technical expert of hydroelectric power plants																																									1				
59	Separator, filter and distiller operator																																			1	1		1				1			
60	Operator of crushing, scraping and polishing machines																																					1		1			1			
61	Head of transport and trucking machines													1																															1	1		
62	Truck and trailer driver																																												1	1		
63	Light truck driver																																												1	1		
64	Crane and dig operator																																														1	
65	machine or dredger operator																																													1	
66	Drilling and loading machines and lethal machines operator																																														1	1
67	Underground mine loading machine operator																																														1	1
68	elevator and crane wheel machine operator																																														1	1
69	operator of trucks and industrial tractors																																														1	1
70	operator of gas compressor equipments and gas pumping stations																																														1	1
71	Operator of oil well pumps																																															

Table 9

The achieved cells for the case study in mode (I)

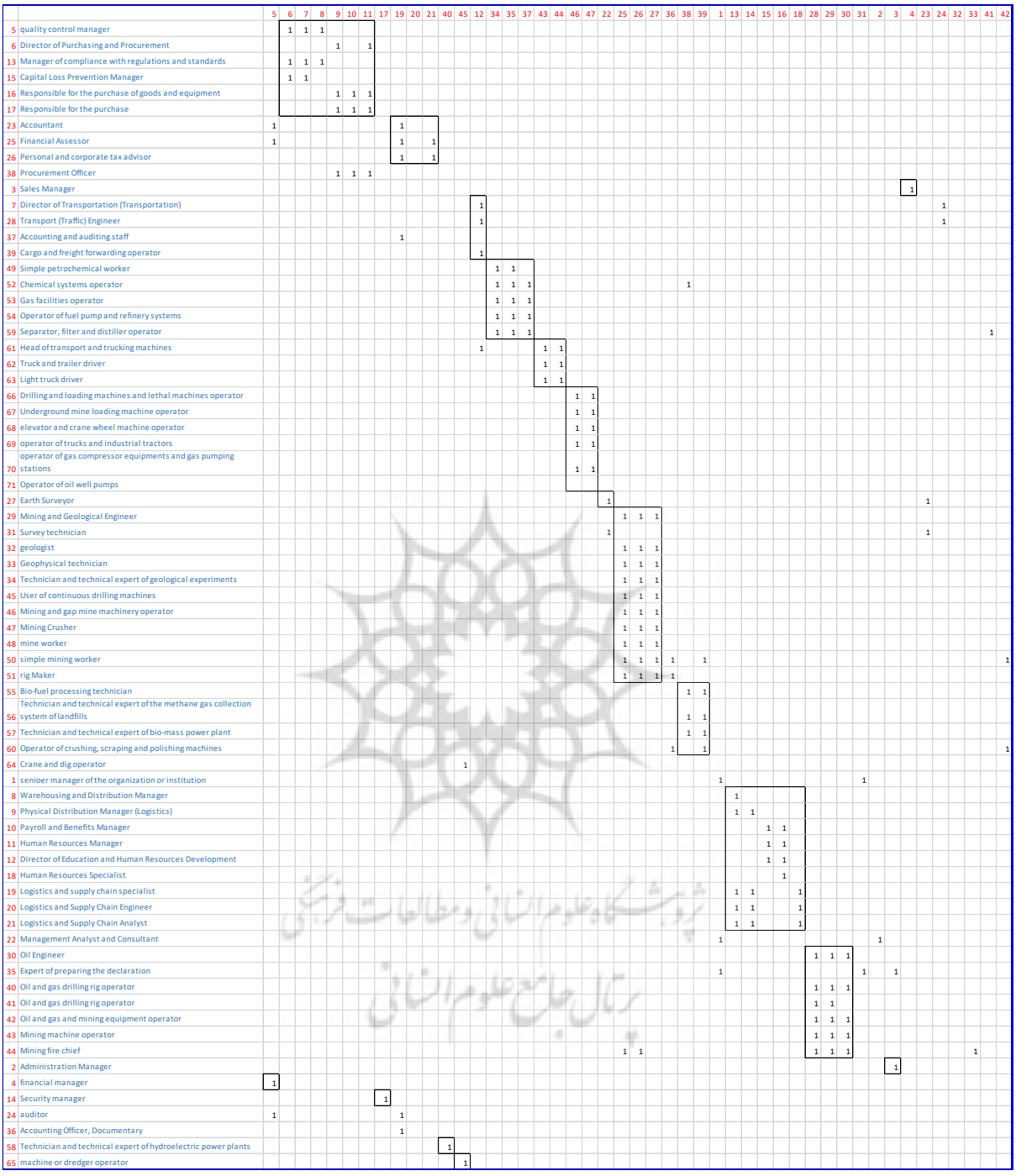
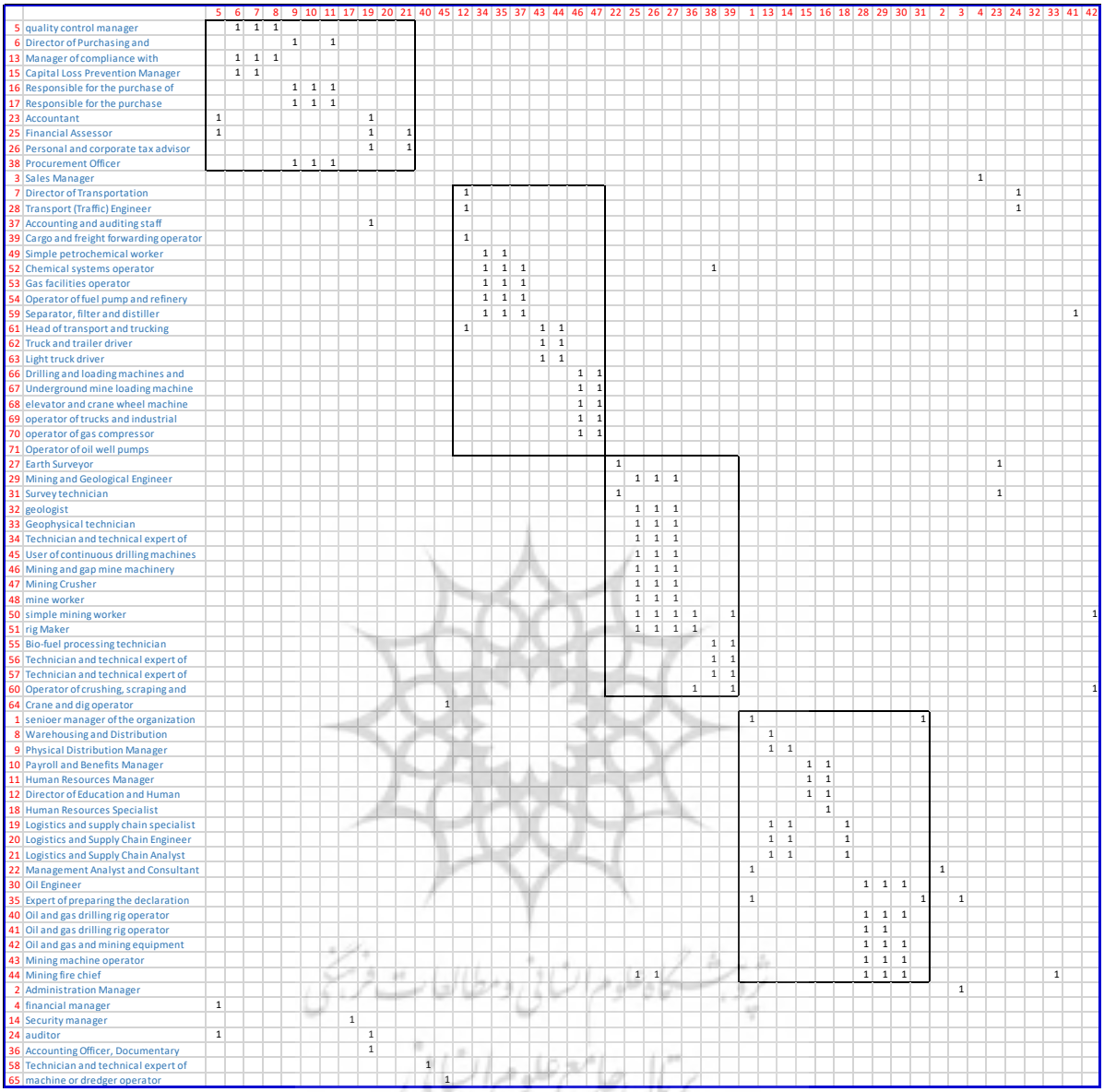


Table 10

The determined cells for the case study in mode (II)



6. Advantages of proposed approach

We know that organizations use traditional methods of job classification, i.e., grouping individuals into larger groups that are then used for personnel-related functions. In traditional methods for grouping jobs, the analysts often employ descriptive evaluation methods. This approach is applied using decision-making meetings such as brainstorming, brainwriting, nominal group technique (NGT) and focus groups. In descriptive evaluation, practitioners discuss several important considerations such as the type of jobs, similarities between jobs, job requirements, moral aspects, and spiritual needs. Indeed, they try to make a reasonable connection between job requirements and human characteristics such as skill, education, age, creativity, motivation, and concentration. Therefore, compared with descriptive evaluation, the proposed quantitative approach, i.e., the application of CFP, has several advantages:

- It is easier to understand and implement and does not require any additional interpretations.
- It avoids cognitive biases. Opposite to descriptive evaluation, the proposed approach is not subject to several biases such as belief bias, actor-observer bias, anchoring bias, availability heuristic, and bandwagon effect.
- It is appropriate for any number of jobs and persons. Descriptive evaluation would be ineffective for big-size problems, while the proposed approach does not have such limitations.

7. Conclusions

This paper proposed an HSA to solve the CFP problem. Afterward, the parameters of the HSA were tuned using a design of experiments (DOE): Taguchi. Moreover, an experimental evaluation and a comparison study between the HSA and existing results in the literature were performed. The results indicated and confirmed good performance and reasonable runtime of the proposed HSA.

The paper also discussed the wide range of skills in megaprojects of the oil and gas industry and concluded that grouping of human resources (called job grouping) was a necessary action. The paper showed that this situation of job grouping was exactly the CFP. In line with this idea, a real-life case study of job grouping in a megaproject was taken, and the application of the proposed HSA to solve it was reported. We assert that the need for the proposed model originated from a real-world project, so it could be employed in future similar actual challenges.

The research innovation included two aspects: (I) developing an HSA for solving the CFP and (II) matching the problem of job grouping with the well-known CFP, which has not been studied yet.

This paper takes a simple real-life case that considers only skills with employees. We know that, in general, a given industrial situation may need to consider several aspects in addition to skills, e.g., skill levels, multiple or cross skills based on similarity of skills, experience, wage levels, aspirations, career progression, and many more features, to group employees. Future models can be expanded to allow for additional features for job grouping in megaprojects. Moreover, this paper was an effort to match the job grouping problem with CFP. This idea may strike future research to extend the notion grouping of processes according to the technologies and equipment as well as the classification of projects according to their similar attributes. Another area for future research belongs to the CFP algorithms. The applicability of the CFP is often limited due to the unavailability of an interactive software program. Hence, it would be beneficial if future methods were developed to include interactive support software for facilitating industrial applications.

Funding

This study was not funded by any organization.

Conflict of interest

The authors declare that they have no conflict of interest.

Ethical approval

This article does not contain any studies with human participants or animals performed by any of the authors.

Nomenclature

ABC	Artificial bee colony
ACO	Ant colony optimization
B&B	Branch and bound
BFA	Bacteria foraging algorithm
BGEP	Bi-cluster graph editing problem
BW	Band width
BWO	Black widow optimization
CFOPT	Cell formation optimization
CFP	Cell formation problem
CSA	Clonal selection algorithm
CSO	Cat swarm optimization
DOE	Design of experiments
E&P	Exploration and production
EMA	Electromagnetism algorithm
GA	Genetic algorithm
GWO	Gray wolf optimization
HAS	Harmony search algorithm
HBBA	Human behavior-based algorithm
HM	Harmony memory
HMCR	Harmony memory considering rate
HMS	Harmony memory size
ICA	Imperialist competitive algorithm
ILS	Iterated local search
LCA	League championship algorithm
MaxIt	Maximum of the iteration number
NA	Not-assigned
NGT	Nominal group technique
PAR	Pitch adjusting rate
PSO	Particle swarm optimization
RPD	Relative percentage deviation
S/N	Signal-to-noise
SA	Simulated annealing
SS	Scatter search
TS	Tabu search
VND	Variable neighborhood descent
WFA	Water flow-like algorithm

References

Arkat, J., Abdollahzadeh, H., and H. Ghahve. (2012). A new branch and bound algorithm for the Cell Formation Problem. Applied Mathematical Modelling, 36(10), 5091–5100.

Arunagiri, A., Marimuthu, U., Gopalakrishnan, P., Slota, A., Zajac. J., and Paulraj, M.P. (2017). Sustainability formation of machine cells in group technology systems using modified artificial bee colony algorithm. Sustainability, 10(1), P. 1–21.

Askin, R., and Subramanian, S. (1987). A cost-based heuristic for group technology configuration. International Journal of Production Research, 25(1), 101–113.

Ayough, A., and Khorshidvand, B. (2019). Designing a manufacturing cell system by assigning workforce. Journal of Industrial Engineering and Management, 12(1), 13–26.

Bagheri, M., and Bashiri, M. (2014). A hybrid genetic and imperialist competitive algorithm approach to dynamic cellular manufacturing system. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 228(3), 458–470.

Behnia, B., Mahdavi, I., Shirazi, B., and Paydar, M.M. (2019). A bi-level bi-objective mathematical model for cellular manufacturing system applying evolutionary algorithms. Scientia Iranica, 26(4), 2541–2560.

Boctor, F. (1991). A linear formulation of the machine part cell formation problem. International Journal of Production Research, 29(2), 343–356.

Boe, W., and Cheng, C. (1991). A close neighbor algorithm for designing cellular manufacturing systems. International Journal of Production Research, 29(10), 2097–2116.

Borghei, S.H., Teymourian, E., Mobin, M., Komaki, G.M., and Sheikh, S. (2015). Enhanced imperialist competitive algorithm for the cell formation problem using sequence data. International Journal of Industrial and Manufacturing Engineering, 9(10), 3532–3541.

Brusco, M. J., (2015). An iterated local search heuristic for cell formation, Computers and Industrial Engineering, 90, 292–304.

Bychkov, I., Batsyn, M. and Pardalos, P.M. (2014). Exact model for the cell formation problem. Optim Lett, 8, 2203–2210.

Carrie, A. (1973). Numerical taxonomy applied to group technology and plant layout. International Journal of Production Research, 11(4), 399–416.

Chan, H., and Milner, D. (1982). Direct clustering algorithm for group formation in cellular manufacture. Journal of Manufacturing Systems, 1, 65–75.

Chandrasekharan, M., and Rajagopalan, R. (1986). An ideal seed non-hierarchical clustering algorithm for cellular manufacturing. International Journal of Production Research, 24(2), 451–464.

Chandrasekharan, M., and Rajagopalan, R. (1989). GROUPABILITY: An analysis of the properties of binary data matrices for group technology. International Journal of Production Research, 27(6), 1035–1052.

Chandrasekharan, M., and Rajagopalan, R. (1987). ZODIAC: An algorithm for concurrent formation of part-families and machine-cells. International Journal of Production Research, 25(6), 835–850.

Chandrasekharan, M.P. and Rajagopalan, R. (1986). MODROC: an extension of rank order clustering for group technology. International Journal of Production Research, 24(2), 1221–1233.

Danilovic, M., and Ilic, O. (2019). A novel hybrid algorithm for manufacturing cell formation problem. Expert Systems with Applications, 135, 327–350.

Dehnavi-Arani, S., Sadegheih, A., Mehrjerdi, Y.Z., and Honarvar, M. (2020). A new bi-objective integrated dynamic cell formation and AGVs' dwell point location problem on the inter-cell unidirectional single loop. Soft Computing, 24(21), 6021–6042.

Elbenani, B., Ferland, J.A. and Bellemare, J. (2012). Genetic algorithm and large neighborhood search to solve the cell formation problem. Expert Systems with Applications, 39, 2408–2414.

Figueroa-Torrez, P., Duran**,** O., Crawford, B., Cisternas-Caneo, F**.** (2023). A binary black widow optimization algorithm for addressing the cell formation problem Involving alternative routes and machine reliability. Mathematics, 11, 3475.

Flyvbjerg, B. (2017). The Oxford Handbook of Megaproject Management. Oxford University Press. ISBN 978–0198732242.

Geem, Z.W., Kim, J.H., and Loganthan, G.V. (2001). A new heuristic optimization algorithm: harmony search. Simulation, 76(2), 60–68.

Goncalves, J., and Resende, M. (2004). An evolutionary algorithm for manufacturing cell formation. Computers and Industrial Engineering, 47(2-3), 247–273.

Hashemoghli, A., Mahdavi, I., and Tajdin, A. (2019). A novel robust possibilistic cellular manufacturing a model considering worker skill and product quality. Scientia Iranica, 26(1), 538–556.

Hatefi, M.A., and Balilehvand, H.R. (2023). Risk assessment of oil and gas drilling operation: An empirical case using a hybrid GROC-VIMUN-modified FMEA method. Process Safety and Environmental Protection, 170, 392–402.

Hazarika, M. (2023). An improved genetic algorithm for the machine-part cell formation problem. International Journal of System Assurance Engineering and Management, 14, 206–219.

Kamalakannan, R., and Pandian, R.S. (2018). A tabu search strategy to solve cell formation problem with ratio level data, International Journal of Business Intelligence and Data Mining, 13(1), 1–3, P. 209–220.

Kamalakannan, R., Pandian, R.S., and Sivakumar, P. (2019). A simulated annealing for the cell formation problem with ratio level data, International Journal of Enterprise Network Management, 10(1), P. 78–90.

Kao, Y., and Lin, C. H. (2012). A PSO-based approach to cell formation problems with alternative process routings. International Journal of Production Research, 50(15), 4075–4089.

Karoum, B., and Elbenani, Y.B. (2017). A clonal selection algorithm for the generalized cell formation problem considering machine reliability and alternative routings. Prodia Engineering, 11, 545–556.

King, J. (1980). Machine-component grouping in production flow analysis: An approach using a rank order clustering algorithm. International Journal of Production Research, 18, 213–232.

King, R., and Nakornchai, V. (1982). Machine-component group formation in group technology: Review and extension. International Journal of Production Research, 20(2), 117–133.

Kumar, C.S. and Chandrasekharan, M.P. (1990). Group efficacy: a quantitative criterion for the goodness of block diagonal forms of binary matrices in group technology. International Journal of Production Research, 28(2), P. 233–243.

Kumar, K., and Vannelli, A. (1987). Strategic subcontracting for efficient disaggregated manufacturing. International Journal of Production Research, 25(12), P. 1715–728.

Kumar, K., Kusiak, A., and Vannelli, A. (). Grouping of parts and components in flexible manufacturing systems. European Journal of Operations Research, 24, 387–397.

Kusiak, A., and Cho, M. (1992). Similarity coefficient algorithm for solving the group technology problem. International Journal of Production Research, 30(11), 2633–2646.

Kusiak, A., and Chow, W. (1987). Efficient solving of the group technology problem. Journal of Manufacturing Systems, 6(2), 117–124.

Li, X., Baki, M.F., and Aneja, Y.P. (2010). An ant colony optimization meta-heuristic for machine-part cell formation problems. Computers and Operations Research, 37(12), 2071–2081.

Liu, C., Wang, J., Leung, J., and Li, K. (2015). Solving cell formation and task scheduling in cellular manufacturing system by discrete bacteria foraging algorithm, International Journal of Production Research, 54(3), 923–944.

Martins, I.C., Pinheiro, R.G. S., Protti, F., and Ochi, L.S. (2015). A hybrid iterated local search and variable neighborhood descent heuristic applied to the cell formation problem. Expert Systems with Applications, 42, 8947–8955.

McCormick, W., Schweitzer, P., and White, T. (1972). Problem decomposition and data reorganization by a clustering technique. Operations Research, 20, 993–1009.

Mosier, C., and Taube, L.A. (1985). The facets of group technology and their impact on implementation. OMEGA, 13(5), P. 381–391.

Mosier, C., and Taube, L.B. (1985). Weighted similarity measure heuristics for the group technology machine clustering problem. OMEGA, 13(6), 577–578.

Nagaraj, G., Arunachalam, M., and Vinayagar, K. (2020). Enhancing the performance of cell formation problem using hybrid efficient swarm optimization. Soft Computing, 24, 16679–16690.

Noktehdan, A., Seyedhosseini, S., and Saidi-Mehrabad, M. (2016). A metaheuristic algorithm for the manufacturing cell formation problem based on grouping efficacy. International Journal of Advanced Manufacturing Technology, 82(1), P. 25–37.

Pailla, A., Trindade, A.R., Parada, V., and Ochi, L.S. (2010). A numerical comparison between simulated annealing and evolutionary approaches to the cell formation problem. Expert Systems with Applications, 37(7), P. 5476–5483.

Pinheiro, R.G.S., Martins, I.C., Protti, F., Ochi, L.S., Simonetti, L.G., and Subramanian, A. (2016). On solving manufacturing cell formation via Bicluster Editing. European Journal of Operational Research, 254, 769–779.

Rabbani, M., Keyhanian, S., Manavizadeh, N., and Farrokhi-Asl, H. (2017). Integrated dynamic cell formation-production planning: A new mathematical model. Scientia Iranica, 24(5), 2550–2566.

Rafee, M., and Mohamaditalab, A. (2020). Investigation into skill leveled operators in a multi-period cellular manufacturing system with the existence of multi-functional machines. Scientia Iranica, 27(6), 3219–3232.

Rostami, A., Paydar, M.M., and Asadi-Gangraj, E. (2020). Dynamic virtual cell formation considering new product development. Scientia Iranica, 27(4), 2093–2107.

Saeidi, S., and Nikakhtar, N. (2022). A revised model for solving the cell formation problem and solving by gray wolf optimization algorithm. Journal of Industrial Engineering and Management Studies, 9(7), 81–94.

Seifoddini, H. (1989). A note on the similarity coefficient method and the problem of improper machine assignment in group technology applications. International Journal of Production Research, 27(7), 1161–1165.

Seifoddini, H., and Wolfe, P. (1986). Application of the similarity coefficient method in group technology. IIE Transactions, 18(3), 271–277.

Soto, R., Crawford, B., Gonzalez, F., Vega, E., Castro, C., and Paredes, F.B. (2019). Solving the manufacturing cell design problem using human behavior-based algorithm supported by autonomous search. IEEE Access, 7, 132228–132239.

Soto, R., Crawford, B., Toledo, A.A., De La Fuente-Mella, H., Castro, C., Paredes, F., and Olivares, R. (2019). Solving the manufacturing cell design problem through binary cat Swarm optimization with dynamic mixture ratios. Computational Intelligence and Neuroscience, 4787856.

Spiliopoulos, K., and Sofianopoulou, S. (1998). An optimal tree search method for the manufacturing systems cell formation problem, European Journal of Operational Research, 105, 537–551.

Srinivasan, G., Narendran, T., and Mahadevan, B. (1990). An assignment model for the part families problem in group technology. International Journal of Production Research, 28(1), P.145–152.

Stanfel, L. (1985). Machine clustering for economic production. Engineering Costs and Production Economics, 9, 73–81.

Tavakkoli-Moghaddam, R., Ranjbar-Bourani, M., Amin, G.R., and Siadat, A. (2012). A cell formation problem considering machine utilization and alternative process routes by scatter search. Journal of Intelligent Manufacturing, 23(4), 1127–1139.

Tavanayi, M., Hafezalkotob, A., and Valizadeh, J. (2021). Cooperative cellular manufacturing system: A cooperative game theory approach. Scientia Iranica, 28(5), 2769–2788.

Waghodekar, P., and Sahu, S. (1984). Machine-component cell formation in group technology MACE. International Journal of Production Research, 22, 937–948.

Wei, N.C., Lin, H.K., and Wu, P. (2012). An electromagnetism-like mechanism for solving cell formation problems. Scientific Research and Essays, 7(9), 1022–1034.

Wu, T., Chung, S., and Chang, C. (2010). A water flow-like algorithm for manufacturing cell formation problems. European Journal of Operational Research, 205, 346–360.

Xue, G., and Offodile, O.F. (2020). Integrated optimization of dynamic cell formation and hierarchical production planning problems. Computers and Industrial Engineering, 139, 106155.