Asian Journal of Information Technology 15 (17): 3287-3295, 2016
ISSN: 1682-3915

A Discrete Particle Swarm Optimization for Cellular Manufacturing System

[1]R. Kamalakannan, [2]R. Sudhakara Pandian, [3]T. Sornakumar and [4]S.S. Mahapatra
[1]Department of Mechanical Engineering, Vickram College of Engineering, Madurai, 630 56 Tamil Nadu, India
[2]Department of Mechanical Engineering, SACS MAVMM Engineering College, Madurai, 625 301 Tamil Nadu, India
[3]Department of Mechanical Engineering, Thiagarajar College of Engineering, Madurai, 625 015 Tamil Nadu, India
[4]Department of Mechanical Engineering, National Institute of Technology, Rourkela, 769 008 Orissa, India

Abstract: Group Technology (GT) is a helpful approach to expand productivity with a high caliber in cell manufacturing frameworks in which cell development is a key stride to the GT theory. The cell development problem is considered as a major issue by many of the investigators who utilize binary machine part occurrence matrix that is formed by the course sheet in the cell manufacturing system. The ones that are present in the binary matrix symbolize the visit of the components to the corresponding machines and the zeros that are represented as components of non-visit. The present study addresses the problem of assembling the cell development through the Discrete Particle Swarm Optimization (DPSO) algorithm. Particle Swarm Optimization (PSO) is a population-based evolutionary algorithm that approaches a social manner of the swarm. The condition used to cluster the machines and components in cells is based upon the minimization of exceptional elements and voids. In this study, we utilized the permutation predicated representation for the encoding scheme for particle position representation. The proposed algorithm performance is verified over the issues that are formed from the open literature and the results that are obtained is then compared with that of benchmark issues which are established from the literature.

Key words: Cellular manufacturing, part-machine grouping, particle swarm optimization, investigators, established

INTRODUCTION

These days, the manufacturing system ought to have the capacity to create items with low manufacturing cost and good quality in time. Simultaneously, manufacturing systems ought to have the adaptability rapid changes in product design and demand with low speculation. So as to achieve high profitability in the batch production systems under the clamorous generation environment, impelling of the cellular manufacturing was intentional (Car and Mikac, 2006). Group Technology (GT) is an assembling philosophy that tries to distinguish and group comparative parts to exploit their resemblance in assembling and design (Irani, 1999). Cell Manufacturing (CM) is an associate application of the GT concepts to production line reconfiguration and shop floor layout design. For actualizing better assembling format the cell development dilemma is considered as a critical problem since it makes a difference from the traditional layouts (Vitanov *et al*., 2007). The part of cell assembling will definitely broaden not just for enhancing efficiency during a typical batch sort producing systems but conjointly for correct adoption of computer-aided manufacturing systems, lean producing frameworks. Exact application of cellular assembling can result the advantages for instance, the process of work in inventory, time set up is reduced, handling of materials, machine usage increased, operator productivity and utilization (NagendraParashar, 2009).

The cell development priority spot the machine cells from parts families in which movement of components from cell to different cell is minimum. To promote this, Machine Part Occurrence Matrix [MPOM] is built under the guidance of the course cards from all the elements. In MPOM the row acts as machines; column as parts and the entry which as the binary values ‘one’ and ‘zero’. Here

Corresponding Author: R. Kamalakannan, Department of Mechanical Engineering, Vickram College of Engineering, Madurai, 630 56 Tamil Nadu, India

the value one acts as the part interrelated to the specific column which is to be deal with the machine related to the particular row and ‘zero’ represents otherwise. MPOM is Block Diagonalized to capitulate ones along the diagonal blocks. Machines and part family are separated from a MPOM and form a diagonal blocks of the matrix that contains gradually ones and fewer zeros. On the off-chance that there are any ones in the off diagonal block, it signifies the inter-cell movement of the concerned parts which are known as exceptional elements (Venkumar and NoorulHaq, 2005). Various methods are there in the literature to acquaint the Cell assembling complications, specifically Non-hierarchical and hierarchical cluster, array manipulation, mathematical programming, meta-heuristics and heuristics, graph theory exist (Schaller, 2005). It is found that this technique generates a greater result for a well-structure matrix with part families that exist and machine cells. However, they fail in the block diagonalization for ill-structured matrixes where it ends up with many exceptional elements.

The cell assembling issue is a combinatorial optimization dilemma that is NP-hardness. Metaheuristics are emerging to solve this dilemma with a worldwide or a near global optimum at a sensible computational time. Boctor (Boctor, 1991) developed a linear formulation of the machine-part cell formation problem. Here, the study has made a suggestion that new zero-one linear formulation which is a new function is removed the interference obtained from the various models. Venugopal and Narendran (1992) proposed genetic algorithm is considered as more flexible to solve problems of cell formation with different targets. It was demonstrated that the proposed algorithm can be an effective tool to gain employment in a cellular based manufacturing environment. Brown and Sumichrast (2011) opted to the grouping of a genetic algorithm for the problem of cell formation. Mahapatra and Sudharkarapandian (2008) considered the sequence of operations and the operational time of the machine parts necessary for processing in the machines. With this, they attempted to develop a genetic algorithm is considered as a combination of the objective by minimizing the variation in cell load and the elements that are expected. Sudhakarapandian and Mahapatra (2009) took their effort towards the manufacturing cell formation using the production data by means of a neural network.

PSO is considered as a metaheuristic approach and it is a population-based, has been endeavored for tackling distinctive engineering problems, recently. Sha & Hsu [33] aimed to propose a PSO algorithm the scheduling of job shop problems by applying preference list-based representation. They proved that this PSO approach is better than other existing PSO methods. Andres and Lozano (2006) introduced the first PSO algorithm for manufacturing cell formation issues. Duran *et al*. (2010) originated from PSO algorithm collaborative model through the technique of data mining for the manufacturing of cell design. Kashan *et al*. (2014) observed a DPSO algorithm for the problem of cell formation by planning the genetic algorithm perception. Analyzing the literature review reveals that there are a number of cellular manufacturing problems solved using several Metaheuristics algorithm. Despite there’s an admittance to utilize PSO algorithm to solve the cellular manufacturing problems, after all, there prevail attempts to use PSO algorithm. In this studyr, a PSO algorithm based metaheuristic approach to solving the cell formation issues is commenced. This projected method is built on the problem of discrete optimization. The achievement of this algorithm is systematically evaluated in comparison with benchmark issues.

MATERIALS AND METHODS

Problem description: In order to create the machine cells and part families, a machine-part occurennce matrix of about 0-1 has been provided, the proper alignment of columns and rows is done in the task of cell formation.In this paper, to determine a rearrangement an attempt is made so that the elements that are exceptional is reduced and the use of machines in a cell can be increased. One acknowledged way to deal with the cell formation the incidence matrix of the machine part is in the cell diagonal creation and to exchange the column and rows in order to propose that the ones towards the diagonal. In this one cell is represented by considering each and every block of diagonal. Cell formation problem targets on the assign of the parts and machine of the same cell if the part visits the machine for processing or assign the machine and part to different cells. At that point, there will be an impeccable cell formation, on the off chance that it fulfills the above objectives.

Figure 1a shows a 6×8 machine part incidence matrix related to the problem with six machines and eight parts. For example, part P1 has operations on Machine M2 and M3. M1, M4 and M6 are not required for processing part P1. Whereas part P2 goes to machine M1 and M2 for its operation and M3,M4,M5& M6 do not take any operation for P2 and thus, it goes on for all the remaining parts. On the off chance, the layout of the machines is done as like Figure 1a. There exists a grouping problem for the smooth flow of parts into the machines. Hence, there is a demand for a novel approach to deal with the formation of cells that leads to an uninterrupted flow of parts in the cellular

(a)	P1	P2	P3	P4	P5	P6	P7	P8
M1	0	1	1	0	0	0	0	0
M2	1	1	1	1	1	0	0	0
M3	1	0	0	1	0	0	1	0
M4	0	0	0	0	1	1	1	1
M5	0	0	1	0	1	0	0	1
M6	0	0	0	0	0	1	1	0

(b)	P5	P6	P7	P8	P1	P2	P3	P4
M4	1	1	1	1	0	0	0	0
M5	1	0	0	1	0	0	1	0
M6	0	1	1	0	0	0	0	0
M1	0	0	0	0	0	1	1	0
M2	1	0	0	0	1	1	1	1
M3	0	0	1	0	1	0	0	1

Fig. 1: a) Machine part incidence matrix; b) block diagonalized matrix

manufacturing systems. After grouping of parts, the goodness of the cell formation problem has to be quantified. Obtaining the diagonalized cell formation as an output, it is shown in Fig. 1b containing two diagonal cells. Chandrasekharan and Rajagopalan (1986) established the first measure in grouping efficiency to validate the occurrence of block diagonal structure generated by cell formation method. The efficiency grouping (η) is the total average weight of two factors (η1 and 2). The 'w' is considered as a weighting factor which lies among 0 to 1 and the resulting value is based upon the matrix size. There occurs a drawback in efficiency grouping with minimized capability of discrimination which is nothing but the capability of differentiating the bad solution and good solution. To overcome these short inadequacies in grouping efficiency Sureshkumar and Rajagopalan (1990) proposed a grouping efficacy (τ). The efficacy grouping is not disturbed by the matrix size. The study has considered the efficacy grouping for the measure of performance in to evaluate the proposed PSO approach:

$$\eta = (w \times \eta_1) + (w \times \eta_2) \quad (1)$$

$$\eta_1 = \frac{\text{Sum of ones in the diagonal matrix}}{\text{Sum of elements in the diagonal matrix}} \quad (2)$$

$$\eta_2 = \frac{\text{Sum of zeros in the off-diagonal matrix}}{\text{Sum of elements in the off-diagonal matrix}} \quad (3)$$

$$t = \frac{(1-\psi)}{(1-\phi)} \quad (4)$$

$$\psi = \frac{\text{Sum of exceptional elements}}{\text{Sum of operations}} \quad (5)$$

$$\phi = \frac{\text{Sum of voids in the diagonal matrix}}{\text{Sum of operations}} \quad (6)$$

There is the formulation of the difficult mathematical machine-component problem of grouping (Boctor, 1991). The model of optimization is as follows. Let us consider, M as the total number of machines, P as the total number of parts, C as the total number of cells, i as the machine index, i.e., i = 1, 2, …, M, j as the part of incidence, i.e., j = 1, 2, ….., P, k as the cell index, i.e., k = 1, 2, …, C and M-X-P is considered as a machine-part occurrence for binary. The intention is selected in order to reduce the time which gives the machine part which can be handled by the machine which does not belong to the cell. The machine part is dispensed to:

$$x_{ik} = \begin{cases} 1, & \text{if machine } i \in \text{cell } k \\ 0, & \text{otherwise} \end{cases}$$

$$y_{jk} = \begin{cases} 1, & \text{if part } i \in \text{family } k \\ 0, & \text{otherwise} \end{cases}$$

$$z_{ijk} = \begin{cases} 1, & \text{if } x_{ik} \neq y_{jk} \\ 0, & \text{otherwise} \end{cases}$$

The problem is symbolized by the following mathematical model. Minimize:

$$Z = \text{Min} \frac{1}{2} \sum_{i=1}^{M} \sum_{j=1}^{P} \sum_{k=1}^{C} a_{ij} Z_{ijk} \quad (7)$$

Subject to:

$$\sum_{K=1}^{C} X_{ik} = 1 \quad \forall i \quad (8)$$

$$\sum_{K=1}^{C} X_{ik} = 1 \quad \forall i \quad (9)$$

$$Z_{ijk} \geq X_{ik} - Y_{jk} \quad (10)$$

$$Z_{ijk} \geq Y_{jk} - X_{ik} \quad (11)$$

$$X_{ij} = Y_{jk} = Z_{ijk} = 0.1 \quad (12)$$

The function objective is given as the number of times which occurs when the machine part needs and it don't appropriate towards the issue of the cell with the parts of family and to the parts where they are allocation. Equation 8 and 9 corresponds to other machines that are allocated to only one cell in this the part of each cell is assigned to one to one family and the Eq. 10 and 11 represent that if machine i and part j are in different cells then Z_{ijk} will take the value as 1.

Proposed discrete particle swarm optimization algorithm for CMS: As per the study by Kennedy and Eberhart (1995), there is an innovative computation technique known as PSO Algorithm which got inspiration through the societal manner of bird flocking or fish schooling. This algorithm had its origin from the social psychological theory and identified as assisting in addressing engineering issues involving multiple optima, non-linearity and on-differentiability and high dimensionality by means of implementation. PSO is primarily population-based search algorithm and its initiation takes place through population having random solutions, known as particles. Hence, position and velocity are the two attributes taken into account to represent a particle. Whenever the need is to alter the particle's position, it is executed through the application of information on the previous position and its related velocity. Each particle was aware of its suitable position and the perfect position attained in the group. These principles are subjected to formulation in terms of:

$$V_k^{t+1} = c_1 * V_k^t + c_2\left(P_{k\,(Best)}^t - X_k^t\right) + c_3\left(G_{k\,(Best)}^t - X_k^t\right) \quad (13)$$

$$P_k^{t+1} = P_k^t + V_k^{t+1} \quad (14)$$

where, c_1-c_3 are learning coefficients. There are three items in the equation's right-hand side (Eq. 13). The previous velocity of the agent becomes the first term. In order to alter the agent's velocity, the need is to apply second and third terms. The tendency of the agent is that it flies in the same direction till it collide the boundary, provided there are no second and third terms. The purpose here is to find out new zones where the first term remains in alignment with diversification as a part of exploring methodology. However, if there is no first term, current position and the best position are the two characters instrumental in ascertaining the velocity of the 'flying' agent. The attempts especially on the part of agents involving the method of searching are making the convergence of their personal bests and global bests. Hence, the terms remain in context to the way search procedure gets intensified. There are three features responsible for governing PSO such as evaluation, comparison and imitation. The evaluation phase takes care of the way each particle assists in addressing the problem. On the other hand, the comparison phase ensures identification of the best particles. Finally, the imitation phase is responsible for generating new particles on the basis of the best particles identified before. Till the occurrence of ceasing criterion, these three phases keeps on repeating. The objective, after all, is to identify the particle that would be perfect in solving the targeted problem. Conventional PSO Algorithm was developed for continuous domains. A DPSO Algorithm was expounded by Kennedy and Eberhart (1995).

Basic elements of discrete particle swarm optimization algorithm: The fundamental parts of PSO Algorithm are as follows:

Particle: X_k^t indicates the k^{th} particle in the swarm at iteration 't' and is denoted by 'n' number of elements as $X_k^t = [X_{k1}^t, X_{k2}^t, X_{k3}^t, \ldots, X_{kn}^t]$ where, X_{kj}^t is the position value of k^{th} particle with respect to the jth element (j = 1, 2, .., n).

Population: In the population, pop^t is the set of NP particles in the swarm at iteration t, i.e., $pop^t = [X_1^t, X_2^t, X_3^t, \ldots, X_k^t]$.

Machine-part sequence: A variable β_k^t which is the machine-part order sequence of the particle X_k^t. It can be described as $\beta_k^t = [\beta_{k1}^t, \beta_{k2}^t, \ldots, \beta_{kn}^t]$, where β_{kj}^t is the machine part order sequence j of the particle k at iteration t.

Particle velocity: V_k^t is the velocity of particle k at iteration t. It can be defined as $V_k^t = [V_{k1}^t, V_{k2}^t, V_{k2}^t, \ldots V_{kn}^t]$, V^{kj} is the velocity of particle k at iteration t with respect to the jth element.

Local best: $P_{k\,(Best)}^t$ denotes the best position of the particle k with the best fitness until iteration t, so the best position related with the best fitness value of the particle k obtained so far is called the local best.

Global best: The $G_{k\,(Best)}^t$ denotes the best position achieved in the group among all the particles k.

Procedural steps of the proposed discrete PSO: The procedure for executing the discrete PSO Algorithm is denoted by step by step.

Step 1: Initialize a swarm of particles, $X_k^t = [X_{ki}^t, X_{k2}^t, X_{k2}^t, \ldots X_{kn}^t]$, machine-part sequences as $\beta_k^t = [\beta_{k1}^t, \beta_{k2}^t, \ldots, \beta_{kn}^t]$ with the random velocities and positions in the problem space.

Step 2: For each particle, calculate the minimized fitness value $Z(X_k^t)$.

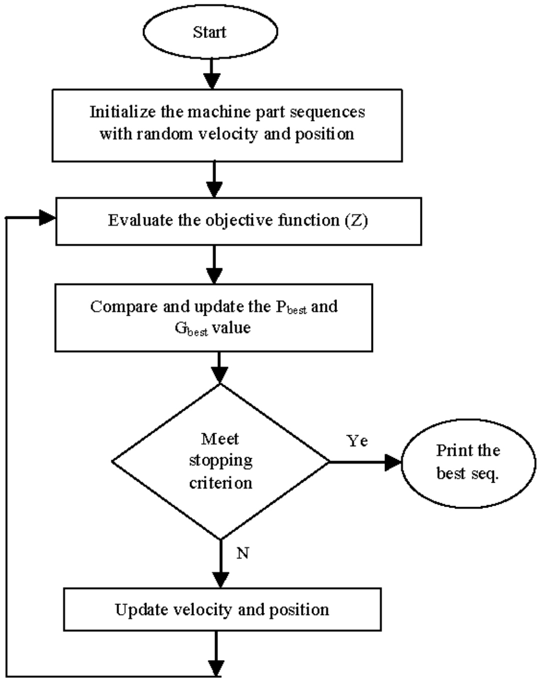

Fig. 2: PSO process

Step 3: Compare the fitness value $Z(X_k^{t+1})$ with its previous best $Z(X_k^t)$; if the current value is better than the previous best and then set the previous best equal to the current value: $P_{k\,(Best)}{}^t = X_k^{t+1}$. This is P_{Best}.

Step 4: Pinpoint the particle in the neighbourhood with the best achievement so far and store it as $G_{k(Best)}$. This is G_{Best}.

Step 5: Apply the local search algorithm to all the particles at the end of the each iteration and calculate the minimized fitness value $Z(X_k^t)$.

Step 6: Modify the particle velocity using their Eq. 13.

Step 7: Modify the particle position using their Eq. 14.

Step 8: Go to step (2) until a Number of iterations is met (t_{max}).

PSO process: The proposed PSO process is shown in the flowchart as in Fig. 2.

RESULTS AND DISCUSSION

Numerical illustration of DPSO algorithm: In Table 1 represents the matrix of machine incidence of an instance problem (6×8) where eight indicates that eight parts are being processed with the use of six machines.

Table 1: Machine part incidence matrix (6×8)

Variable	P1	P2	P3	P4	P5	P6	P7	P8
M1	1	1	0	0	1	0	0	1
M2	0	0	1	1	0	1	1	0
M3	1	1	0	0	1	1	0	1
M4	0	0	0	1	0	0	1	1
M5	0	0	1	1	0	1	1	0
M6	1	1	0	0	0	1	0	0

For a better understanding, Let us consider the following data during one particular 'tth' iteration for the above problem. In this, the string size is taken as the sum of machines and parts for the initial iteration the current position and best position of each particle are same. In the succeeding iterations the best and current position will change for all the particles, but updating during iteration is global best:

$$P_{Current} = X_k^t = \{M1, M2, M3, M4, M5, M6, P7, P8, P9, P10, P11, P12, P13\,\&\,P14\}$$

$$P_{Best} = P_{k(Best)}^t = \{M1, M2, M3, M4, M5, M6, P7, P8, P9, P10, P11, P12, P13\,\&\,P14\}$$

$$G_{Best} = G_{k(Best)}^t = \{M6P7P8P9P10P11P12P13\ P14M1M2M3M4M5\}$$

The calculation procedure for the particle velocity and particle movement is as:

Velocity index generation: The velocity index of a particle k at iteration t is computed using Eq. 15:

$$V_k = (i_u, j_u); u = 1, L \tag{15}$$

In Eq. 15 i and j represent the machine part position and L indicates the list length. Let us consider L_k as velocity length has to equal to 14 (machines+parts) with the randomly generated list:

$$V_k^t = \{(12,13)(14,1)(2,3)(4,5)(6,7)(8,9)(10,11)(5,9)(7,4)(3,9)(4,7)(8,5)(3,6)(5,3)\}$$

Similarly the velocity index will be generated for the other remaining particles.

Calculation of particle velocity: Using Eq. 13 the velocity of particles that are moving from one place to another is calculated for each particle in the population. In the learning value of the coefficient, the velocity operator is used to find the velocity number and the components are implied to the position. For instance, the coefficient value of learning is 0.5 then, it is 50% of the components of velocity that are selected towards the list of velocity and the position applied.

Calculation of particle velocity: The particle is moving from their current position to the new position using Eq. 14:

$$P_k^{t+1} = \begin{Bmatrix} M1, M2, M3, M4, M5, M6, P7, P8, \\ P9, P10, P11, P12, P13\& \ P14 \end{Bmatrix} + \{(12,13)(14,1)(2,3)(4,5)(6,7)(8,9)(10,11)(1,6)(2,7)(3,8)(4,9)(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} M1, M2, M3, M4, M5, M6, P7, P8, \\ P9, P10, P11, P13P12, \&P14 \end{Bmatrix} + \{(14,1)(2,3)(4,5)(6,7)(8,9)(10,11)(1,6)(2,7)(3,8)(4,9)(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P14, M2, M3, M4, M5, M6, P7, P8, \\ P9, P10, P11, P13P12, \&M1 \end{Bmatrix} + \{(2,3)(4,5)(6,7)(8,9)(10,11)(1,6)(2,7)(3,8)(4,9)(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P14, M3, M2, M4, M5, M6, P7, P8, \\ P9, P10, P11, P13P12, \&M1 \end{Bmatrix} + \{(4,5)(6,7)(8,9)(10,11)(1,6)(2,7)(3,8)(4,9)(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P14, M3, M2, M5, M4, M6, P7, P8, \\ P9, P10, P11, P13P12, \&M1 \end{Bmatrix} + \{(6,7)(8,9)(10,11)(1,6)(2,7)(3,8)(4,9)(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P14, M3, M2, M5, M4, P7, M6, P8, \\ P9, P10, P11, P13P12, \&M1 \end{Bmatrix} + \{(8,9)(10,11)(1,6)(2,7)(3,8)(4,9)(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P14, M3, M2, M5, M4, P7, M6, P9, \\ P8, P10, P11, P13P12, \&M1 \end{Bmatrix} + \{(10,11)(1,6)(2,7)(3,8)(4,9)(5,10)(6,11)(7,12)\}$$

Table 2: Output machine part incidence matrix for the problem (6×8)

Variable	P1	P2	P3	P4	P5	P6	P7	P8
M3	1	1	1	1	1	0	0	0
M6	1	0	0	1	1	0	0	0
M5	1	1	1	1	1	0	0	0
M4	0	0	1	0	0	1	0	1
M1	0	0	0	0	0	1	1	1
M2	0	0	0	0	1	1	1	1

$$= \begin{Bmatrix} P14, M3, M2, M5, M4, P7, M6, P9, \\ P8, P11, P10, P13P12, \&M1 \end{Bmatrix} + \{(1,6)(2,7)(3,8)(4,9)(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P7, M3, M2, M5, M4, P14, M6, P9, \\ P8, P11, P10, P13P12, \&M1 \end{Bmatrix} + \{(2,7)(3,8)(4,9)(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P7, M6, M2, M5, M4, P14, M3, P9, \\ P8, P11, P10, P13P12, \&M1 \end{Bmatrix} + \{(3,8)(4,9)(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P7, M6, P9, M5, M4, P14, M3, M2, \\ P8, P11, P10, P13P12, \&M1 \end{Bmatrix} + \{(4,9)(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P7, M6, P9, P8, M4, P14, M3, M2, \\ M5, P11, P10, P13P12, \&M1 \end{Bmatrix} + \{(5,10)(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P7, M6, P9, P8, P11, P14, M3, M2, \\ M5, M4, P10, P13P12, \&M1 \end{Bmatrix} + \{(6,11)(7,12)\}$$

$$= \begin{Bmatrix} P7, M6, P9, P8, P11, P10, M3, M2, \\ M5, M4, P14, P13P12, \&M1 \end{Bmatrix} + \{(7,12)\}, \ P_k^{t+1} = \begin{Bmatrix} P7, M6, P9, P8, P11, \\ P10, P13, M2, M5, M4, \\ P14, M3P12, \&M1 \end{Bmatrix}$$

Correspondingly for all the subdivisions P_k^{t+1} is calculated and $P_{Current}$, P_{Best} and , P_{Best} are updated. Table 2 shows the output for the above 6x8 problem after little iteration. There are two machine cells and two part families. From this it can observed that there are two exceptional elements for P6 & P8 from the corresponding M2 &M4, respectively. In the first cell having machines {M3, M6, M5} and parts {P2, P5, P8, P1 and P6} and the second cell having machines {M4, M1 and M2} and parts {P7, P3 and P4}. The exceptional element is the element which occurs outside the diagonal block due to machines and parts are allocated to two different cells. In Table 2 there are two exceptional elements, i.e., $a_{26} = a_{48} = 1$. The voids are the elements which occur inside the diagonal block showing that there is no processing between the corresponding machine and part. In Fig. 1b there are three voids i.e., $a_{65} = a_{68} = a_{43} = 0$. Several Performance measures has been developed for quantifying the goodness of the cell formation problem. The grouping efficacy for this cell formation problem is calculated using the Eq. 4 and it is measured as 80.77%

RESULTS AND DISCUSSION

In order to examine the performance of PSO, there are several set of experiments have been conducted. Parameter tuning of PSO algorithm is done in order to get the best solution. The results that are obtained is then compared with the other existing algorithms found in the literature. The DPSO algorithm which is proposed is coded in C^{++} and it is tested under i3 processor which is under Windows 7.

Parameter tuning: In all the metaheuristics, parameter tuning is a vital feature to control the objective function as needs be. Here sensitivity examination is utilized to locate the best parameter esteem for the proposed DPSO algorithm. The optimal size of the swarm is identified by solving the issue of cell formation by a various set of problems for the objective by considering the decreasing the elements exceptional by adapting PSO algorithm.The derived swarm size 40 is a good result that has been discovered and the same has been used all through the assessment. In this proposed work, the termination criterion is preferred as 10,000 iterations for this single objective optimization issue. The important parameters of the coefficient learning are c1, c2 and c3 that is formed by each and every particle towards the gbest solution and local best solutions all along the search process. This, on the other hand, knows as acceleration constants. Low learning coefficient's value will bring about strolling a long way from the objective, i.e., gbest and local best. The coefficient of high learning will bring about the convergence of the premature search process. To pinpoint the best suitable value of learning coefficient's combinations, the sensitivity analysis is conducted. The results of the analysis are shown in Fig. 3. The target capacity esteem Z which is present in PSO reduces when there is an increase towards the coefficient learning and

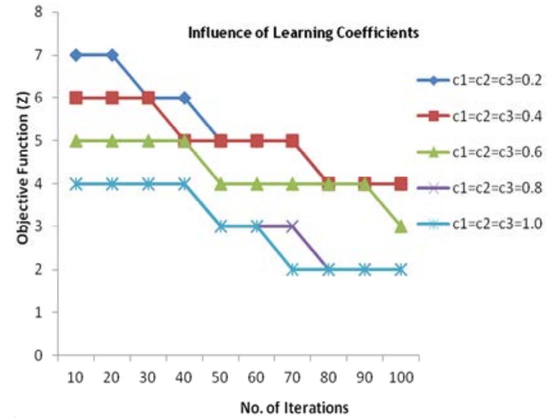

Fig. 3: Effect of PSO Learning Coefficients c1-3 on objective funciton (Z)

the state of enduring is obtained after particular interactions for the output machine part occurrence matrix for the problem (6×8).

Benchmark problems: It is found that there are various methods have been proposed from the literature review for the problem of cell formation.The performances used by them were examined in different instances in terms of different performance measures. In this work, the grouping efficacy is used as the performance measure for DPSO algorithm. The proposed PSO results are then compared with the other benchmark problem that is found in the literature review.

Computational results: With the above parameters of the PSO, we apply and report the results to compare our algorithm with a method that existing in the literature review. The problems which it is considered are the selected matrices having the dimensions from 5×7-20×35 with the well and ill-structured matrices. The experiments have defined to choose a large iteration number as 10000 and 40 as a number of particles. The algorithms which we have chosen for the comparisons are listed as follows: ZODIAC, GRAFICS, MST-Clustering Algorithm, GATSP, GA & EA. Since in this, the first three algorithms which we describe do not allow the presence of singletons, i.e., cells containing only one machine or one part. In our problem, we do not consider the singletons and residual cells. Table 3 shows the comparison results of PSO and other six algorithms for the sixteen problems. From the computational results, it is clearly observed that this PSO algorithm performs better for the small size problems and solutions are empowering for the larger size problems. By considering the multi-objective, i.e. as minimizing the

Table 3: Comparison of PSO with results from literature

	Grouping efficacy in other approaches					Proposed PSO grouping efficacy					
Source	Size	ZODIAC	GRAFICS	MST	GATSP	GA	EA	Min.	Avg.	Max.	Avg. time(s)
King and Nakornchai (1982)	5×7	73.68	73.68	-	-	-	73.68	73.68	73.68	73.68	0.109
Waghodekar and Sahu (1984)	5×7	56.52	60.87	-	-	62.50	62.50	62.50	62.50	62.50	0.109
Seifoddini (1989)	5×18	77.36	-	-	77.36	77.36	79.59	77.36	77.36	77.36	0.230
Kusiak (1992)	6×8	76.92	-	-	76.92	76.92	76.92	76.92	76.92	76.92	0.164
Kusiak and Chow (1987)	7×11	39.13	53.12	-	46.88	50.00	53.13	39.13	53.13	53.13	0.301
Boctor(1991)	7×11	70.37	-	-	70.37	70.37	70.37	67.90	67.90	67.90	0.301
Seifoddini and Wolfe (1986)	8×12	68.30	68.30	-	-	-	68.30	68.29	68.29	68.29	0.325
Chandrasekharan *et al.* (1989a)	8×20	85.24	85.24	85.24	85.24	85.24	85.25	56.70	56.70	56.70	0.420
Chandrasekharan *et al.* (1989b)	8×20	58.33	58.13	58.72	58.33	55.91	58.72	60.00	60.00	60.00	0.420
Mosier and Taube (1985a)	10×10	70.59	70.59	70.59	70.59	72.79	70.59	70.59	70.59	70.59	0.353
Chan and Milner (1982)	10×15	92.00	92.00	92.00	92.00	92.00	92.00	92.00	92.00	92.00	0.420
Askin and Subramaninan (1987)	14×23	64.36	64.36	64.36	-	-	69.86	45.00	45.00	45.00	0.620
Stanfel (1985)	14×24	65.55	65.55	--	67.44	63.48	69.33	65.90	65.90	65.90	0.630
McCormick *et al.* (1972)	16×24	32.09	45.52	48.70	-	-	52.58	45.52	45.52	48.70	0.789
Mosier and Taube (1985b)	20×20	21.63	38.26	--	37.12	-	42.94	37.12	37.12	37.12	0.890
Carrie (1973)	20×35	75.14	75.14	75.14	75.28	66.30	76.22	65.6	65.6	75.14	0.982

exceptional elements and minimizing the cell load variation till which will enhance the outcome of this proposed method and for the larger sized issues.

CONCLUSION

The above study reveals that DPSO algorithm is developed in order to find a solution to the problem of cell formation. In this PSO, the permutation-based representation is utilized as the encoding scheme for the particle position representation. The proposed algorithm is tested on the issue of the benchmark and the results are then compared with the algorithms that existing. These results show the proposed PSO performs better for the small sized problems and solutions are encouraging for the larger sized problems. Therefore, it is recommended to continue the research in the following directions. Hybridization of the PSO algorithm with other algorithms for solving NP-hardness problems, solving multi-objective optimization problems, finally for other real-time problems by using this PSO algorithm there is a demand to develop the model to specify the optimal number of cells and optimal production mix.

REFERENCES

Abdi, K., M. Fathian and E. Safari, 2012. A novel algorithm based on hybridization of artificial immune system and simulated annealing for clustering problem. Int. J. Adv. Manuf. Technol., 60: 723-732.

Agrawal, A.K., P. Bhardwaj and V. Srivastava, 2011. On some measures for grouping efficiency. Int. J. Adv. Manuf. Technol., 56: 789-798.

Andres, C. and S. Lozano, 2006. A particle swarm optimization algorithm for part-machine grouping. Rob. Comput. Integr. Manuf., 22: 468-474.

Arora, P.K., A. Haleem and M.K. Singh, 2013. Recent development of cellular manufacturing systems. Sadhana, 38: 421-428.

Boctor, F.F., 1991. A Jinear formulation of the machine-part cell formation problem. Int. J. Prod. Res., 29: 343-356.

Brown, E.C. and R.T. Sumichrast, 2001. CF-GGA: A grouping genetic algorithm for the cell formation problem. Int. J. Prod. Res., 39: 3651-3669.

Car, Z. and T. Mikac, 2006. Evolutionary approach for solving cell-formation problem in cell manufacturing. Adv. Eng. Inf., 20: 227-232.

Chandrasekaran, M.P. and R. Rajagopalan, 1987. ZODIAC-an algorithm for concurrent formation of part-families and machine-cells. Int. J. Prod. Res., 25: 835-850.

Chandrasekharan, M.P. and R. Rajagopalan, 1986. An ideal seed non-hierarchical clustering algorithm for cellular manufacturing. Int. J. Prod. Res., 24: 451-464.

Chandrasekharan, M.P. and R. Rajagopalan, 1986. MODROC: An extension of rank order clustering for group technology. Int. J. Prod. Res., 24: 1221-1233.

Duran, O., N. Rodriguez and L.A. Consalter, 2010. Collaborative particle swarm optimization with a data mining technique for manufacturing cell design. Expert Syst. Appl., 37: 1563-1567.

Elbenani, B., J.A. Ferland and J. Bellemare, 2012. Genetic algorithm and large neighbourhood search to solve the cell formation problem. Expert Syst. Appl., 39: 2408-2414.

Goncalves, J.F. and M.G. Resende, 2004. An evolutionary algorithm for manufacturing cell formation. Comput. Ind. Eng., 47: 247-273.

Irani, S.A., 1999. Handbook of Cellular Manufacturing Systems. John Wiley & Sons, New York, USA.,.

James, T.L., E.C. Brown and K.B. Keeling, 2007. A hybrid grouping genetic algorithm for the cell formation problem. Comput. Oper. Res., 34: 2059-2079.

Jeffrey, S.E., 2005. Tabu search procedures for the cell formation problem with intra-cell transfer costs as a function of cell size. Comput. Ind. Eng., 49: 449-462.

Kao, Y. and S.C. Fu, 2006. An ant-based clustering algorithm for manufacturing cell design. Int. J. Adv. Manuf. Technol., 28: 1182-1189.

Kashan, A.H., B. Karimi and A. Noktehdan, 2014. A novel discrete particle swarm optimization algorithm for the manufacturing cell formation problem. Int. J. Adv. Manuf. Technol., 73: 1543-1556.

Kennedy, J. and R. Eberhart, 1995. Particle swarm optimization. Proc. IEEE Int. Conf. Neural Networks, 4: 1942-1948.

King, J.R. and V. Nakornchai, 1982. Machine-component group formation in group technology: Review and extension. Int. J. Prod. Res., 20: 117-133.

King, J.R., 1980. Machine-component grouping in production flow analysis: An approach using a rank order clustering algorithm. Int. J. Prod. Res., 18: 213-232.

Kumar, C.S. and M.P. Chandrasekharan, 1990. Grouping efficacy a quantities criterion for goodness of block diagonal forms of binary matrices in group technology. Intl. J. Prod. Res., 28: 223-243.

Kusiak A., 1992. The generalized group technology concept. Int. J. Prod. Res., 25: 561-569.

Kusiak, A. and H. Lee, 1996. Neural computing-based design of components for cellular manufacturing. Int. J. Prod. Res., 34: 1777-1790.

Li, X., M.F. Baki and Y.P. Aneja, 2010. An ant colony optimization metaheuristic for machine-part cell formation problems. Comput. Oper. Res., 37: 2071-2081.

Mahapatra, S.S. and R.S. Pandian, 2008. Genetic cell formation using ratio level data in cellular manufacturing systems. Int. J. Adv. Manuf. Technol., 38: 630-640.

Noktehdan, A., B. Karimi and A.H. Kasha, 2010. A differential evolution algorithm for the manufacturing cell formation problem using group based operators. Expert Syst. Applied, 37: 4822-4829.

Onwubolu, G.C. and M. Mutingi, 2001. A genetic algorithm approach to cellular manufacturing systems. Comput. Ind. Eng., 39: 125-144.

Pandian, R.S. and S.S. Mahapatra, 2009. Manufacturing cell formation with production data using neural networks. Comput. Ind. Eng., 56: 1340-1347.

Papaioannoua, G. and J.M. Wilson, 2010. The evolution of cell formation problem methodologies based on recent studies (1997-2008): Review and directions for future research. Eur. J. Operat. Res., 206: 509-521.

Parashar, B.N., 2009. Cellular Manufacturing Systems-An Integrated Approach. PHI Learning Private Limited, New Delhi, India,.

Rameshkumar, K., R.K. Suresh and K.M. Mohanasundaram, 2005. Discrete Particle Swarm Optimization (DPSO) algorithm for permutation flowshop scheduling to minimize makespan. Proceedings of the International Conference on Advances in Natural Computation, August 27-29, 2005, Changsha, China, pp: 572-581.

Sayadi, M.K., A. Hafezalkotob and S.G.J. Naini, 2013. Firefly-inspired algorithm for discrete optimization problems: An application to manufacturing cell formation. J. Manuf. Syst., 32: 78-84.

Sha, D.Y. and C.Y. Hsu, 2006. A hybrid particle swarm optimization for job shop scheduling problem. Comput. Ind. Eng., 51: 791-808.

Srinivasan, G. and T.T. Narendran, 1991. GRAFICS-a nonhierarchical clustering algorithm for group technology. Int. J. Prod. Res., 29: 463-478.

Srinivasan, G., T.T. Narendran, and B. Mahadevan, 1990. An assignment model for the part-families problem in group technology. Int. J. Prod. Res., 28: 145-152.

Srinivasana, G., 1994. A clustering algorithm for machine cell formation in group technology using minimum spanning trees. Int. J. Prod. Res., 32: 2149-2158.

Venkumar, P. and A.N. Haq, 2005. Manufacturing cell formation using modified ART1 networks. Int. J. Adv. Manuf. Technol., 26: 909-916.

Venugopal, V. and T.T. Narendran, 1992. A genetic algorithm approach to the machine-component grouping problem with multiple objectives. Comput. Ind. Eng., 22: 469-480.

Vitanov, V., B. Tjahjono and I. Marghalany, 2007. A decision support tool to facilitate the design of cellular manufacturing layouts. Comput. Ind. Eng., 52: 380-403.