Meta-heuristics in cellular manufacturing: A state-of-the-art review

Tamal Ghosh[a*], Sourav Sengupta[a] , Manojit Chattopadhyay[b] and Pranab K Dan[a]

[a]*Department of Industrial Engineering,& Management, West Bengal University of Technology, BF 142, Salt Lake City, Kolkata 700064 India*
[b] *Department of Computer Application, Pailan College of Management & Technology, Bengal Pailan Park, 7000104, West Bengal, India*

A R T I C L E I N F O	A B S T R A C T
	Meta-heuristic approaches are general algorithmic framework, often nature-inspired and designed to solve NP-complete optimization problems in cellular manufacturing systems and has been a growing research area for the past two decades. This paper discusses various meta-heuristic techniques such as evolutionary approach, Ant colony optimization, simulated annealing, Tabu search and other recent approaches, and their applications to the vicinity of group technology/cell formation (GT/CF) problem in cellular manufacturing. The nobility of this paper is to incorporate various prevailing issues, open problems of meta-heuristic approaches, its usage, comparison, hybridization and its scope of future research in the aforesaid area.

1. Introduction

Cellular manufacturing (CM) has been evolved to fulfil contemporary market demand where traditional manufacturing system was incompetent. Therefore, CM is a solution to efficient batch type with low setup time to produce variety of part types, shorter lead time and higher machine utilization with superior quality (Sudhakarapandian, 2007). Group technology (GT) is defined as a technique which distinguishes similar parts and clustering them into part families based on their manufacturing designs, attributes and geometric shapes and it was first proposed by Burbidge (1963). GT is applied in cellular manufacturing as an alternative of traditional manufacturing system. Designing manufacturing cell is usually called cell formation problem (CF/CFP) which consists of the following approaches: similar parts are normally grouped into part families according to their processing requirements, dissimilar machines are grouped to form manufacturing cells and consequently part families are allocated to cells. Depending on the procedures involved in CFP, three solution methodologies are proposed by Selim et al. (1998): (a) part families are accomplished first and hence machines are clustered into cells according to the processing requirement of part families. This is known as part-family identification, (b) manufacturing cells (clustering of heterogeneous machines) are first generated based on uniformities in part routing and then the part families are allocated to

cells. This is known as machine groups' identification, (c) part families and machine cells are formed concurrently, which is known as part families/machine grouping.

Despite the fact that there have been large number of solution methodologies proposed by researchers since early 80s to solve CF problems, such as mathematical programming, graph theory, exact methods, heuristics, meta-heuristic methods and artificial intelligent techniques such as neural network and fuzzy set theory, the clear research trend in literature of CFP (Papaioannou & Wilson, 2010) manifests a direction towards soft-computing methodologies due to its strong nature of converging to attain optimal solution. Meta-heuristic which is a sub-branch of soft computing, exclusively evolutionary algorithms, tabu search, simulated annealing, ant colony optimization, particle swarm optimization, bees algorithm, water flow-like algorithm are the frequently adopted techniques of this class, and being employed by researchers in CFP in search of better solution promptly. For a better understanding, the notation of this survey, Table 1 summarizes all the necessary abbreviations used in this paper.

Table 1
List of abbreviations used in this study

Abbreviations	
NP: *Non Polynomial*	ANOVA: *Analysis of Variance*
GT: *Group Technology*	MOGGA: *Multi-Objective Grouping Genetic Algorithm*
CM: *Cellular Manufacturing*	VSM : *Volume Sensitivity Model*
CMS: *Cellular Manufacturing System*	MGA : *Modified Genetic Algorithm*
CFP: *Cell Formation Problem*	ART: *Adaptive Resonance Theory*
TS: *Tabu Search*	NSGA II: *Non-Dominated Sorting Genetic Algorithm II*
EA: *Evolutionary Algorithm*	IAECLP: *Intra-cell And Inter-Cell Layout Problem*
ACO: *Ant Colon Optimization*	DECF: *Differential Evolution Cell Formation*
PSO: *Particle Swarm Optimization*	EnGGA : *Enhanced Grouping Genetic Algorithm*
BA: *Bees Algorithm*	HMA-RTM: *Hybrid Memetic Algorithm and Revised TOPSIS method*
WFA: *Water Flow-like Algorithm*	
SA: *Simulated Annealing*	SPEA-II: *Strength Pareto Evolutionary Algorithm II*
GA: *Genetic Algorithm*	MOSS : *multi objective scatter search*
TSCF: *Tabu Search Cell Formation*	WIP: *Work in Progress*
GAA: *Group And Assign Method*	ACS : *Ant Colony System*
TSH: *Tabu Search Heuristic*	TSP: *Travelling Salesman Problem*
CBTSH: *CB Tabu Search Heuristic*	VCMS : *Virtual Cellular Manufacturing System*
SCFP : *Sustainable Cell Formulation Problem*	ACC : *Ant Colony Clustering*
MOTS: *multi-objective tabu search*	FPSO: *Fuzzy Particle Swarm Optimization*
CSDP: *Cellular System Design Problem*	QPSO: *Quantum Particle Swarm Optimization*
EEs: *Exceptional Elements*	HSAM: *Hybrid Simulated Annealing with Mutation*
SAHCF: *Simulated Annealing Heuristic Cell Formation*	HGA: *Hybrid Genetic Algorithm*
TSHCF: *Tabu Search Heuristic Cell Formation*	PSA: *Parallel Simulated Annealing*
2D SA: *Two Dimensional Simulated Annealing*	BIP: *Binary Integer Programming*
LP: *Linear Programming*	QAP: *Quadratic Assignment Problem*
DCMS : *dynamic cellular manufacturing system*	MIP: *Mixed-Integer Programming*
MFA-SA: *Mean field Annealing-Simulated Annealing*	NLP: *Non Linear Programming*
EP : *Evolutionary Programming*	DS: *Dataset*
GP : *Genetic Programming*	GGA: *Grouping Genetic Algorithm*
DE : *Differential Evolution*	SLCA: *Single Linkage Clustering Algorithm*
SS: *Scatter Search*	GMPG: *General Machine-Part Grouping*
MA : *Memetic Algorithm*	MOMP: *multi objective mathematical programming*
EOG : *Evolutionary Optimization of Granules*	IP: *integer programming*

2. CF solution methods based on meta-heuristics

Classification of CF based meta-heuristic approaches are demonstrated in a taxonomic framework in Fig. 1, and detailed descriptions are given accordingly in next subsections,
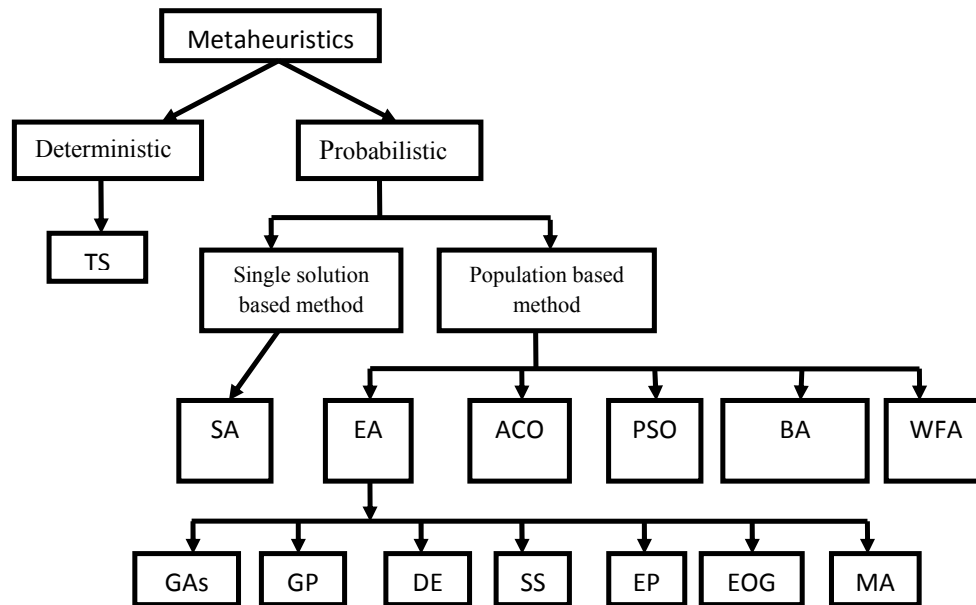

Fig. 1. Taxonomic framework of meta-heuristics

Since cell formation problems are NP-complete in nature (Nair & Narendran, 1999), it is difficult to obtain global solution(s) which leads us to search for near optimal solution(s). Application of meta-heuristics in CFP is emerging which parallels the remarkable ability of mimicking natural or biological phenomena to find 'fittest' solution by incorporating 'survival of the fittest' theory proposed by Darwin (1929). These techniques have the capabilities to solve the hardest amongst NP-complete problems called NP-hard and to obtain near-optimal solution. Meta-heuristic techniques constitute evolutionary approaches (EA), simulated annealing (SA), tabu search (TS), ant colony optimization (ACO), particle swarm optimization (PSO), bees algorithm (BA), water flow-like algorithm (WFA). Since late 90s the applications of meta-heuristic techniques to GT/CF problems have been encouraging. The literature concerning CMS using these major techniques are discussed here.

2.1 Deterministic meta-heuristics

2.1.1 Tabu Search (TS)

Tabu search is believed to be one of the most successful meta-heuristic techniques for the NP-complete applications. A comprehensive introduction to TS can be found in the book by Glover and Laguna (1997). Tabu search is essentially a sophisticated and improved type of local search, an algorithm which in its simplest form, also known as Hill Climbing, works as follows. Consider a starting current solution, evaluate its neighbouring solutions based on a given neighbourhood structure, and set the best or the first found neighbour which is better than the current solution as new current solution and repeat the procedure until an improving solution is detected in the neighbourhood of the current solution. The local search stops when the current solution is better than

all its neighbours, that is, when the current solution is a local optimum. The pseudocode 1 shows the tabu search procedure.

Pseudocode 1: Tabu Search (TS)

---

initialize;
repeat
        generate all of the acceptable neighbourhood solutions;
        evaluate the generated solutions;
        choose the best one as the candidate solution
        if there is no suitable candidate then choose the best of forbidden solutions as the candidate;
        update the tabu list;
        move to candidate solution;
        if the number of generated solutions are sufficient, diversify;
until termination condition is met;

2.1.2 TS in Cell Formation

Logendran et al. (1994) developed CMS design model for selection of machines and unique process plan and hence designed two TS based heuristic each with 2 methods namely method 1 and method 2. They further proposed an extensive statistical analysis based on randomized block design and reported that heuristic 2 had better performance than heuristic 1. Sun et al. (1995) modelled the CFP with an objective of minimizing inter-cell material flows as a graph partition problem and developed a TS-based iterative improvement algorithm to solve the resulted problem. The algorithm improves existing cell configuration through a simple local searching scheme. Aljaber et al. (1997) designed the CFP based on graph theory and a pair of shortest spanning path problems, and proposed a TS heuristic for the solution of the problems, which produced better quality solutions with higher CPU time. Lozano et al. (1999) presented one-step approach to part-machine grouping and he assumed some limits to the sizes of machine cells and part families. He then implemented a TS algorithm which was benchmarked against several SA techniques, heuristics and another TS method and a quadratic integer programming model was proposed with the help of weighted sum of intracell voids and intercell moves, where his proposed method outperformed other procedures with reduced computational time. Onwubolu and Songore (2000) addressed CFP with three objective functions: minimizing intercell moves, minimizing cell load variation and combining both the former objectives and designed a TS method which offers freedom to consider maximum cell size and number of machines within cell and they reported encouraging results. Adenso-Diaz et al. (2001) developed a TS based methodology to solve CFP with a focus on different machine grouping problems. They reported that their proposed method could outperform two SA-heuristic techniques with reasonably less execution time for medium to large problems. Spiliopoulos and Sofianopoulou (2003) developed a multi-stage cell design approach where the primary part was implemented by a TS algorithm, integrated with proper short-term and long-term memory structures. The overall search strategy depicts the benefit of adaptive memory and responsive exploration. Design of experiment was also implemented for tuning the input parameters to detect the near-optimal solutions, efficiently. Logendran and Karim (2003) also considered long-term memory based on minimal frequency to solve CFP, and a TS approach was developed to improve solutions which was initially developed followed by six different versions of it in order to investigate the impact of long term memory and the use of fixed versus variable tabu list sizes. All approaches outperformed the mixed-integer programming model obtaining solutions which are close to optimal in no significant amount of time. Cao & Chen (2004) stated a CFP with fixed charge cost by minimizing the summation of inter-cell material handling cost, cell construction cost and machine related costs using an embedded

optimization procedure to transform the original mixed integer programming model into a pure binary problem, hence applied TS to yield optimal or near optimal solution of the reduced problem. Wu et al. (2004) developed comprehensive TS heuristic which consists of dynamic tabu tenure and a long term memory structure known as TSCF for CFP when process plans for parts and production factors such as production volume and cell size were taken into account. Two other methods for quickly generating the initial solutions were also developed, namely GAA and the random approach. Computational results were observed to be promising for a GAA accompanied with TS approach for small to medium sized problems. Tavakkoli-Moghaddam et al. (2005) explained that dynamic condition of CFP becomes more complex and proposed TS, SA and GA methods to solve this type of problems. Their study indicated that SA is better in terms of solution and complexity than TS, GA, but by improving GA operator's functionalities can also produce better result since this can be added with other meta-heuristic approaches such as TS, SA. Jeffrey Schaller (2005) stated new heuristics based on TS namely TSH, CBTSH for CFP and compared the solution with existing methods from literature. Study depicts although both the above methods are good but CBTSH is recommended due to its ability to handle large problems. Foulds et al. (2006) introduced mixed integer programming model combined with assignment of parts to individual machines, the grouping of individual machines into cells, and the modification of individual machines to increase their part processing capability, called sustainable cell formulation problem (SCFP) heuristic and solved this class of problems with tabu search with much better result. Lei and Wu (2006) worked with multi-objective CF and proposed a Pareto-optimality based on multi-objective tabu search (MOTS) with different objectives: minimization of the weighted sum of intercell and intracell moves and minimization of the total cell load variation. A new approach was stated to determine the non-dominated solutions among the solutions produced by the TS. The computational results demonstrated strong ability of MOTS to find Pareto-optimal solution. Ateme-Nguema and Dao (2007) investigated an ACO based TS heuristic for cellular system design problem (CSDP) and the methodology proved to be much quicker than traditional methods when considering operational sequence, time and cost. Rodrigues and Weller (2008) considered alternative routing to minimize extra-cellular processing of task and a branch and bound based hybrid TS was also designed to solve the CFP and the proposed technique was then compared successfully with the available methods in the literature. Ateme-Nguema and Dao (2009) further proposed quantized Hopfield network for CFP to find optimal or near-optimal solution and TS was employed to improve the performance and the quality of solution of the network. Wu et al. (2009) proposed a hybrid TS to solve CFP and its variants and the core solution searching algorithm combined in the scheme could be easily modified to other meta-heuristic approaches, such as the SA, GA, based on the problem characteristics or the user preferences. This methodology uses mutation operation of GA to avoid early convergence to local optimum.

Preceding study reports the significance of TS based methodologies in cell formation problem; while Table 2 illustrates various frameworks of TS methods.

2.2 Probabilistic meta-heuristics

2.2.1 Single solution based method

Simulated annealing (SA) is found as the only algorithm in this class which is applied on cell formation problems which is the oldest among meta-heuristic methods. The SA algorithm simulates the physical annealing process, where particles of a solid arrange themselves into a thermal equilibrium.

Table 2
Various attributes of proposed TS based methodologies

References	Initial Construction	Neighborhood structure and transition rule	Stopping criteria
Logendran et al. (1994)	Smallest achievable annual operating cost of parts determines initial solution	Single move needed to reach next configuration and Forward perturbation scheme adopted	Specified no. of local optima evaluated or prescribed CPU time lapses
Sun et al. (1995)	Randomly generated	Single or double move needed to reach next set of configurations and move is not forbidden & the move maximizes the gain	prescribed computational time or a prescribed number of transitions performed
Aljaber et al. (1997)	Random or a heuristic solution	Adjacent Pairwise Interchange or insert or swap move proposed.	number of iterations exceeds a specified constant or without improving the current solution
Lozano et al. (1999)	Random generation	Exchange & insertion move for machines and union and splitting move for cells	number of iterations without significant improving the current solution
Onwubolu and Songore (2000)	machines are randomly assigned to cells	Feasible transfer of one machine from one cell to another. Intensification and diversification employed to improve the search	the intensification and diversification lengths used to terminate the solution search
Adenso-Diaz et al. (2001)	Random generation	Exchange, insertion, union and splitting moves	number of iterations exceeds a specified constant or without improving the current solution
Spiliopoulos and Sofianopoulou (2003)	Random generation	Simple move of machine from cell to cell or swap move of two machines	iterations are stopped when the corresponding value can no more be improved
Logendran and Karim (2003)	specific neighbourhood function used to generate feasible solution	Inside and outside perturbation schemes adopted for machine location identification and part machine assignment	number of iterations without improvement and the number of entries into the inside index list
Cao & Chen (2004)	Random generation	Using swap move neighborhood configuration is generated	predetermined number of iterations has been reached; or the solution has not been improved after a certain number of consecutive iterations
Wu et al. (2004)	Random approach and the group-and-assign method	Single, exchange and double moves are proposed	If the iteration limit is exceeded
Tavakkoli-Moghaddam et al. (2005)	Random generation	Generate neighbouring solution X^n by move m $(X^{n-1} \rightarrow X^n)$	Predefined Number of accepted solutions
Jeffrey Schaller (2005)	a feasible solution consists of an assignment for each operation for each part to a cell	Move is created by assigning the operation of one part to a cell that is different from its assignment and retaining all of the other cell assignments for the operations for each of the parts	If the three tabu list sizes each fail to produce an improved solution
Foulds et al. (2006)	Generated by Initial allocation of machines to cells	single transformation applied with the least objective function value	If best value achieved and doesn't change in consecutive iteration
Lei and Wu (2006)	Stochastically generate an initial feasible solution	Exchange move between stochastically or randomly selected machines	predetermined number of iterations
Ateme-Nguema and Dao (2007)	Cell configuration proposed using ACO	ANT based move using probability for an ant to select an arc between two machines	Error less than a predefined value
Ateme-Nguema and Dao (2009)	iterative process employed	Hybrid Hopfield network determines neighborhood set	when the error is smaller or equal to a fixed threshold value
Wu et al. (2009)	similarity coefficients methods and rank order clustering can generate feasible solution	Mutation operator applied to invoke neighborhood configuration	If best value achieved and doesn't change in consecutive iteration

An introduction to SA can be found in the book by Aarts and Korst (1990). The standard type of applications concerns combinatorial optimization problems of the following form where S is a finite set of feasible solutions.

$$\min_{x \in S} g(x)$$

The algorithm uses a pre-defined neighbourhood structure on 'S'. A control parameter called *temperature* in analogy to the physical annealing process governs the search behaviour. In each iteration, a neighbour solution *y* to the current solution *x* is computed. If *y* has a better objective function value than *x*, the solution *y* is *accepted*, that is, the current solution *x* is replaced by *y*. If, on the other hand, *y* does not have a better objective function value than *x*, the solution *y* is only accepted with a certain probability depending on (i) the difference of the objective function values in *x* and *y*, and (ii) the temperature parameter. The pseudocode 2 demonstrates SA procedure.

Pseudocode 2: Simulated Annealing (SA)

---

initialize;
repeat
      generate a candidate solution;
      evaluate the candidate;
      determine the current solution;
      reduce the temperature;
until termination condition is met;

2.2.2 SA in Cell Formation

Boctor (1991) proposed a mixed-integer linear program based CFP to minimize the number of EEs and employed a SA method which is indeed efficient for small and large-scale experiments by 64%. Venugopal and Narendran (1992) suggested simple SA searching method and applied it on cell design problem in cellular manufacturing which seems to perform better than K-means algorithm for large-scale problem. Liu and Wu (1993) introduced a general form of simulated annealing technique for CFP with due consideration of penalty cost in objective function and reported promising results for some large-size problems. Chen and Srivastava (1994) proposed a quadratic programming model of CFP to maximize the sum of machine similarities within cells, subject to cell size limitation. The proposed SA method shows better performance when compared with graph-partitioning heuristic. Souilah (1995) suggested a SA based resource clustering technique into manufacturing cells and utilize the shop-floor surface effectively and tested the algorithm successfully with numerical examples. Murthy and Srinivasan (1995) introduced fractional CFP model using remainder cell as a linear integer programming problem to minimize count of EEs and proposed a SA and heuristic method. Vakharia and Chang (1997) proposed two combinatorial search approaches for the CF problem based on SA (SAHCF) and TA (TSHCF) for CFP to minimize the total expenditures of the machines and the material handling needed to transfer the loads among cells. The study indicated that SAHCF outperformed TSHCF in terms of solution quality and computational time. Zolfaghari and Liang (1998) considered processing time, machine capacity and machine duplication and a new grouping efficacy which takes into account the processing time and incorporate their SA method. Authors further introduced a Hopfield network for good seed solution and shorter convergence time. Su and Hsu (1998) presented parallel SA for machine-part CFP which minimizes total cost, total machine loading unbalance, also considered operation sequences, setup time, operation time, intercell and intracell transportation cost of a part. The parallel SA uses merits of GA and satisfactory result is obtained while testing on large problems. Zhou and Askin (1998) proposed multiple techniques: a greedy heuristic, minimum increment heuristic, SA heuristic for CFP to minimize machine cost, variable production cost, setup cost and intracell material handling cost and reported good results.

Sofianopoulou (1999) demonstrated a nonlinear integer programming model of CFP by considering processing sequence of each part and developed a 2D SA method to determine machine cells and part-to-process plan assignments and an LP model was developed to find part family and some good results were reported for mid-size problems. Caux et al. (2000) stated a new method to solve cell formation problem with alternative routings and machine capacity constraints. The proposed algorithm simultaneously deals with the cell formation problem and the part-routing assignment problem whereas the other methods are based on branch and bound and SA. One of problems was then solved from the solutions of the other. The method is limited to large-size problem and unconstrained problem due to calculation time. Adil and Rajamani (2000) studied the trade-off between cell compactness and cell independence in terms of cost of intercell and intracell moves and developed a nonlinear mathematical model and SA to minimize the total move costs. Abduelmola and Taboun (2000) implemented productivity model of CFP which was initially formulated as 0-1 integer programming model. They modified SA to solve large-scale problems where input data include the number of parts, machines and cells, demand, selling price, inter and intra-cell costs, and maximum number of machines allowed in each cell. Baykasoglu et al. (2001) proposed multi-objective CFP by minimizing total load imbalance, extra capacity requirement and dissimilarity among parts and formulated a solution methodology based on SA and co-operative game theory approach to handle multi-objectivity. The study shown by Xambre and Vilarinho (2003) is a CFP model with multiple and functionally identical machines to minimize intercell flows by considering flow volume among the operations. Jayaswal and Adil (2004) proposed SA based heuristic methodology for CFP with due consideration of operational sequence, machine replication, alternative process routing to minimize the sum of costs of intercell moves, machine investment and machine operating costs. The algorithm produced good results for large-scale problems. Das et al. (2006) proposed the multi-objective mixed integer-programming model for CMS design by minimizing machine operating and utilization cost and total material handling cost and maximizing system reliability. The methodology introduced is hybridized SA with GA operator to obtain better neighbouring solutions. Mahesh and Srinivasan (2006) addressed a multi-objective incremental CFP and lexicographic based simulated annealing algorithm which yields good results for small-size problems but it depends on initial solution for medium to large-scale problems. Study proposed by Wu et al. (2007) depicted a hybrid SA method with genetic operation considering alternative process routing and insertion move was utilized in solution improvement stage in order to speed up solution search and to escape from local optima. Arkat et al. (2007) developed a sequential CFP model based on SA for large-scale problems and compared their method with GA. They reported similar results for both methods where SA needed less computational time. Safaei et al. (2008) proposed a model of dynamic cellular manufacturing system (DCMS) with different objectives of minimizing total machine cost, intercell and intracell material handling cost, reconfiguration cost and solved their model using mean field annealing (MFA) embedded SA and MFA-SA. This new methodology outperforms conventional SA because of MFA's strong capability to generate initial solution in significant amount of time. Defersha and Chen (2008c) studied a mathematical programming model to form manufacturing cells over multiple time period to minimize different cost components such as machine investment cost, inter-cell material handling cost, operating cost, subcontracting cost, tool consumption cost, setup cost and system reconfiguration cost. They also developed a parallel SA incorporating several problem specific perturbation operators and constraint handling techniques to solve the resulted problem formulation and examined their method on some mid-size problems. Tavakkoli-Moghaddam et al. (2008) introduced an integer programming model for dynamic CFP. A multi-period planning horizon was assumed where product mix and demand were different but deterministic for each period. A SA algorithm was developed and the results were compared with the optimal results found through the mathematical model and reported that the efficiency found with mean deviations from the optimality to be less than 4%. Wu et al. (2008) experimented with a SACF model which is sequential in nature, which follows minimization of number of voids and EEs. This searching technique is guided by single and exchange move in order to converge to optimality. Tavakkoli-Moghaddam et al. (2009) presented common cells and specific cells and part families in such a way that the demand for parts in

each period could be satisfied in a batch size form. In their proposed model there are two kinds of capital constraints: capital constraints to set up cells and capital constraint to provide required equipment to manufacture parts. They also used SA for the proposed model where there are three objectives: Minimization of the sum of costs of delay of delivering the part to the customers by common and specific cells in each period; minimization of the costs of keeping cells idle time for each period; and maximization of the unused capital, to solve. They also compared their results with LINGO 6 software package. A hybrid methodology based on Boltzmann function from simulated annealing and mutation operator from GA was proposed by Wu et al. (2009) to optimize the initial cluster obtained from similarity coefficient method (SCM) and rank order clustering (ROC). The computational experiment shows 36% of the test problems yielded better efficiency measures for CFP. The abovementioned SA based literature survey focuses only on cell formation issues. Therefore, to project the detailed outcomes of individual SA based methodologies and several criteria selection, Table 3a and Table 3b are presented.

2.2.3 Population based methods

Population based methods are those which not only mimic the biological or natural phenomena but also they start with a set of initial feasible solutions called 'population' and the objective would be to guide that search in state space to reach to the optimal solution.

2.2.4 *Evolutionary Approaches (EA)*

Evolutionary algorithms (EAs) are global, parallel, search and optimization methods, found on the principles of natural selection (Darwin, 1929) and population genetics (Fisher, 1930). In general, any iterative, population based approach that uses selection and random variation to generate new solutions can be regarded as an EA. EA is executed iteratively on a set of coded chromosome, called a population, with three basic genetic operators: selection, crossover and mutation. Each member of population is called an individual or a chromosome and is represented by a string. EA uses only the objective function information and probabilistic transition rules for genetic operations. Crossover is the primary operator of EA. The basic structure of an EA algorithm is presented by pseudocode 3. These techniques have its origin in several landmarked evolutionary approaches experimented in CF, mainly seven different categories of EAs are identified, evolutionary programming (EP) (Suer, 1997), genetic programming (GP) (Dimopoulos, 2006), differential evolution (DE) (Kao et al., 2008), scatter search (SS) (Bajestani et al., 2009), memetic algorithm (MA) (Muruganandam et al., 2005), evolutionary optimization of granules (EOG) (Chi and Lin, 2002) and genetic algorithms (GA) (Goldberg, 1989). All these algorithms have the genetic operations embedded inside with minor variations, and other heuristics or meta-heuristics can be combined with these algorithms to form hybrid methods, which are being used in recent literatures. Most heavily adopted algorithm in this category is GA or genetic algorithm.

Pseudocode 3: Evolutionary Approaches (EA)

Initialize;
repeat
        evaluate the individuals;
        repeat
            select parents;
            generate offspring;
            mutate if enough solutions are generated;
        until population number is reached;
        copy the best fitted individuals into population as they were;
Until required number of generations are generated.

Table 3a
Various attributes of proposed SA based methodologies

References	Initial solution	Neighbourhood solution	Temperature reducing function	Stopping condition
Boctor (1991)	generated at random	generated at random	Modified function	Maximum no. of iteration
Venugopal and Narendran (1992)	Randomly assign machines to cells	Randomly swap tow machines	Geometric	Freezing temperature
Chen and Srivastava (1994)	by randomly assigning the m machines into K cells	randomly moving a machine from its present cell to another randomly selected cell	$T_l = T_l / 1 + \lambda\, T_l$	Value of the objective function does not change or number of iterations exceeds the maximum allowed value.
Souilah (1995)	generated at random	generated at random	Modified function taken from literature	a given final temperature is reached
Murthy and Srinivasan (1995)	generated at random	generated at random	Geometric: $T_i = \alpha T_{i-1}$	Maximum iteration (200) or threshold temperature (2.0) value reached
Vakharia and Chang (1997)	a machine and parts assignment to cells	generated at random	Modified function	Best objective value
Su and Hsu (1998)	machines are grouped into cells	Crossover and mutation of GA is used to generate more candidate solution	geometrically decreased with rate 0.95	Freezing temperature
Zhou and Askin (1998)	Heuristic to obtain initial solution	generated at random	geometrically decreased with rate 0.993	$C_k < \varepsilon$
Zolfaghari and Liang (1998)	Generated a random seed solution using improved Hopfield network method.	generated at random by reassigning a machine from its current cell to another cell	$\theta_t = \theta_0 / (1 + \ln t)$	maximum allowed number of iterations
Sofianopoulou (1999)	generated at random	generated at random	geometrically decreased with rate 0.9	Number of iterations exceeds the maximum allowed value.
Caux et al. (2000)	represented by a vector of cells and index no. indicates machine	insertion or a permutation applied	Logarithmic: $T = C/\ln(n+1)$	Number of iterations exceeds the maximum allowed value.
Adil and Rajamani (2000)	The number of cells is set equal to the number of machines.	randomly moving machine to the cell to get new machine assignment	Geometric: $T_i = \alpha T_{i-1}$	Maximum iteration or acceptance ratio reaches its lower bound or objective value does not change
Abduelmola and Taboun (2000)	generated at random	generated at random	Geometric: $T_i = \alpha T_{i-1}$	Best objective value

Table 3b
Various attributes of proposed SA based methodologies

References	Initial solution	Neighbourhood solution	Temperature reducing function	Stopping condition
Baykasoglu et al. (2001)	generated randomly	generated at random	geometric: $T_i = 0.9T_{i-1}$	Maximum iteration or objective value does not change in 200 iterations
Xambre and Vilarinho (2003)	operations are allocated to machines by decreasing order of their usage rate & machines are grouped into cells	Choosing randomly the first core machine & following the initial solution generation steps	geometric: $T_i = 0.9T_{i-1}$	freezing temperature of 10 is set and if no improvement for consecutive 5 temperature level
Jayaswal and Adil (2004)	generated randomly	obtained by perturbing an operation assignment of a part to a different machine type/cell	Geometric: $T_i = \alpha T_{i-1}$	Maximum iteration or acceptance ratio reaches its lower bound or objective value does not change
Das et al. (2006)	Random generation	Crossover and mutation of GA is used to generate more candidate solution	Geometric: $T_i = \alpha T_{i-1}$	Maximum iteration or predefined temperature value reaches
Mahesh and Srinivasan (2006)	Using one of the algorithms developed in past research by authors themselves.	two perturbation schemes with equal probability used.	geometrically decreased with rate 0.95	final temperature is reached
Wu et al. (2007)	Based on routing selection and assignment to machine cells	Mutation or insertion move applied	geometric: $T_i = C.T_{i-1}$	Best objective value found
Arkat et al. (2007)	Random generation	Random generation	geometric: $T_i = C.T_{i-1}$	Freezing temperature
Safaei et al. (2008)	stochastic heuristic is used	Four heuristic operators are used	geometric: $T_i = C.T_{i-1}$ with C ranges between0.85 & 0.95	maximum number of consecutive temperature trials reached
Defersha and Chen (2008c)	Randomly generated	Six different solution perturbation schemes are used	geometric: $T_i = C.T_{i-1}$	Maximum iteration reached
Wu et al. (2008)	using parts assignment and machines assignment procedures	New parts assignment plan through neighbourhood searching by performing single move.	geometrically decreased with rate 0.7	If predefined temperature value reaches
Tavakkoli-Moghaddam et al. (2009)	created in a purely random manner	random value will be uniformly chosen and its corresponding cell will be located	geometric: $T_i = C.T_{i-1}$ with C ranges between0.5 & 0.99	when the temperature will be reached to the required final level
Wu et al. (2009)	Based on routing selection and assignment to machine cells	Mutation or insertion move applied	geometric: $T_i = C.T_{i-1}$	Best objective value found

2.2.5 EA in Cell Formation

Venugopal and Narendran (1992) studied the nature of GA for multi-processor system and efficiently reached to optimality for CFP which deals with multi-objectivity. Gupta et al. (1996) implemented GA as a solution methodology to CF problem and solved multiple objectives such as total movements of components and cell load variation. Joines et al. (1996) developed an integer programming model using GA to solve CFP; the method shows a new chromosome representation which reduces the size of the model, the efficiency was demonstrated by comparing the maximum number of states visited by the GA to the entire state space for sample data sets. Morad and Zalzala (1996) proposed genetic-based methods to solve two problems in the manufacturing systems: the cell-formation problem in CM and the batch scheduling problem. In the cell-formation problem, multi-criteria optimization incorporating processing such as the machine capacity and processing times were used. The results showed that the processing criterion certainly affects the formation of cells. Hwang and Sun (1996) demonstrated a two phase GA heuristic for CFP which was more effective than traditional methods in terms of global efficiency, group efficiency and intercell move factors where cell designers could choose the number of cells and upper limit of cell size. Zhao et al. (1996) introduced fuzzy clustering method for inexact real-data structure and proposed GA due to its population-wide and stochastic nature. Kazerooni et al. (1997) proposed simultaneous grouping of parts into part families and machines into cells by considering production volume, process sequence, alternative routing and developed a GA to solve the problem with greater efficiency. Suer (1997) proposed an evolutionary programming technique for cell formation in cellular manufacturing environment. Al-Sultan and Fedjki (1997) stated a genetic operation based heuristic method and formulated an integer quadratic programming model of CFP and tested against the previously proposed methods with prospective solutions. The approach proposed by Pierreval and Plaquin (1998) is very useful where no prior knowledge is needed to assign weight or a particular distance in the multicriteria problem formulation. The method is based on a niched Pareto evolutionary algorithm. The algorithm shows a set of non-dominated (or Pareto) solutions with respect to several objectives. Gravel et al. (1998) presented a double-loop genetic algorithm which provides a method for computing efficient solutions for the multiple route bicriterion cell formation problems. The method could be implemented to make the best choice of the existing cell design by competent part-routing through the cells. Here only the internal loop of the genetic procedure is used to determine the specific route used for each part. The research work by Hsu and Su (1998) presented a GA which could be effective methodology to group machines when dealing with multiple objectives such as simultaneously minimizing total cost and intracellular and intercellular machine loading imbalances. Moon and Gen (1999) proposed a GA based approach to design independent manufacturing cells by giving due consideration to production volume, machine capacity, processing time, number of cells and cell size. Zhao and Wu (2000) used multiple objectives and part routing of CF problems and solved the resulted model with the help of a modified GA and reported that the method could be time consuming for large-scale problems. Mak and Wong (2000) implemented a CFP model based on total cell flows and a genetic method was also developed for efficient clustering and then ANOVA test was also incorporated to select appropriate system parameters and effectiveness of the technique was demonstrated on some benchmark problems. Mak et al. (2000) suggested a genetic search technique to solve CFP which maximizes bond energy measure. An adaptive scheme was also embedded in the method which helps to adjust the GA parameters while searching and the technique was tested successfully on benchmark problems. Plaquin and Pierreval (2000) developed an evolutionary algorithm based on genetic operators for CFP based on four constraints criterion: bounded size of cells, machines that must stay together, machines that must not stay together, machines around which the cells have to be formed and reported faster convergence characteristics. Lee-Post (2000) proposed that GT coding system (DCLASS) could be efficiently used with SGA to cluster part families which is well suited for part design and process planning in production. The results indicated that the technique could consume negligible computational time to find near-optimal solution. Chu and Tsai (2001) proposed a GA based heuristic technique to model CFP where new similarity coefficient developed to adjust the gene value of each part and heuristic mutation applied to tune the gene value of machine and part. Brown

and Sumichrast (2001) introduced GGA in order to find more efficient solution methodology for machine-part CF problems. Onwubolu and Mutingi (2001) addressed CFP with three objective functions: minimizing intercell moves, minimizing cell load variation and combining both the former objectives and designed a GA method which competed with hybrid GA and TS method. The computational result is indeed encouraging. Dimopoulos and Mort (2001) developed genetic programming (GP) based method to model single linkage clustering (SLCA) problem with multiple objectives. Chi and Lin (2002) proposed new technique called evolutionary optimization of granules (OEG) which is a mixed form of granular computing and GA, applied on CFP, and the result obtained is efficient due to the simplicity of computation and the ability to handle large-scale problem. Wu et al. (2002) proposed a heuristic genetic algorithm with a new dynamic selection method to deal with concurrent decisions which involved highly correlated objectives and a new group mutation operator was developed to increase the mutation probability, to simultaneously solve the cell formation and machine layout decisions, where a two-layer hierarchical chromosome structure was developed for problem domains to deal with concurrent decisions. Zolfagharia and Liang (2003) considered processing time, lot size, and machine capacity for general machine-part grouping (GMPG) problem. They also proposed a GA method where input parameters were carefully tuned using design of experiment and multi-factor ANOVA test. They reported significant improvement and indicated the importance of parameter selection. Mansouri et al. (2003) proposed multi-objective GA to solve multi-objective CF problems; the chromosome is taken here as a vector of many decision variables and the fitness function is a function of multiple sub-objective functions. This tedious technique proposed optimal solution compared with other multi-objective CF methodologies. Zolfaghari and Liang (2004) introduced a GA methodology for CFP which dealt with processing time, lot size, machine capacity, and machine duplication. Solimanpur et al. (2004) introduced a GA with multiple fitness function to solve a multi-objective mathematical programming based model which generates several solutions along the Pareto-optimal frontier and developed decision support system for CF problem. Chi and Yan (2004) attempted to test GA in fuzzy environment considering the manufacturing factors of multi-process plan, fuzzy product demands and fuzzy technical feasibility of machines, the developed methods satisfied for the practical production situations as well as the cellular manufacturing system could become more flexible to match the real application. Chan et al. (2004) proposed a multi-objective mathematical model of machine-part grouping problem with alternative routing, machine aggregation and disaggregation and a GA approach was used to solve the proposed model. According to Goncalves and Resende (2004), GA could be more effective with local heuristics in CFP domain. The research work by Yasuda et al. (2005) showed that GGA was efficient methodology to solve multi-objective CF problem when dealing with processing time, available time on machine. Muruganandam et al. (2005) applied memetic algorithm (MA) which is a modified version of GA embedded with TS on CFP and they reported that MA could outperform when compared with GA and TS individually for large-size problems. A genetic algorithm was used in fuzzy environment by Pai et al. (2005) to solve part-machine CF problem. Vin et al. (2005) introduced a multi-objective grouping genetic algorithm (MOGGA) combined with CF heuristic by considering process sequence, production volume and alternative routing. The evaluation of the solutions was also based on various criteria such as the CF evaluation, the similarity among different products assigned to a machine, the cost and flexibility evaluation on the basis of limit of machine utilization. Rogers and Kulkarni (2005) introduced new method called bivariate clustering of matrix for CFP and a GA based method employed to solve the problem. Rajagopalan and Fonseca (2005) proposed a volume sensitivity model (VSM) for the first time with production volume limit for individual component rather than using product mix and implemented GA model to show that when machine movement is not viable then volume limit can enhance the choice of optimal routing of components. Rajagopalan and Fonseca (2006) further published their GA based model to workout CF problem with an objective to reduce intercellular and intracellular material handling cost with other cost components such as backtracking cost, machine skipping cost and penalty cost. The cost function was developed using heuristic algorithm which was used as fitness function of GA model. The

method is believed to be a significant improvement in cell formation and depicted better grouping efficacy. Filho and Tiberti (2006) introduced grouping genetic algorithm with new crossover, mutation operators, correction scheme and a new codification scheme of chromosomes based on machine groups rather than individual machine and the methodology efficiently seemed to converge faster. Hu and Yasuda (2005) pursued a research based on alternative process routes for cell formation problem and developed a GGA methodology with new chromosome representation, separate crossover heuristic and special mutation technique which produces efficient and optimal solution. Nsakanda et al. (2006) modelled a CFP with multiple dimensions such as operations sequence, part demands, machine capacities, multiple process plans and multiple routings and developed a GA method combined with price-direct decomposition method, and computational experiment produced good results for large-scale problems. Boulif and Atif (2006) stated graph partitioning formulation of CFP which first uses a binary GA and then a branch and bound method to enhance GA. Result produced, the binary GA outperforms classical GA and branch and bound enhanced GA outperforms binary GA. Chan et al. (2006) developed two mathematical models, one was CFP to minimize intracell and intercell part movement and the other was CLP to minimize intercell part travelling distance unit. A GA method was developed for both the problem models to find multiple optimal solutions. Defersha and Chen (2006) developed a mathematical model, which incorporates dynamic cell configuration, alternative routings, sequence of operations, multiple units of identical machines, machine capacity, workload balancing among cells, operation cost, subcontracting cost, tool consumption cost, set-up cost and other practical constraints. A two-phase GA based heuristic technique was also proposed to solve this CFP and the method was tested on some examples with greater efficiency. Wu et al. (2006) introduced a hierarchical GA method to solve CF problem simultaneously with group layout problem. The result shown this concurrent concept is able to produce better quality solution than traditional sequential methods by 2-20%. Car and Mikac (2006) proposed a method to solve CFP based on emergent synthesis idea, which was employed using a modified genetic algorithm (MGA) which is believed to generate better results for CFP problems. Dimopoulos (2006) proposed GP-SLCA model to solve large-scale problems. His technique is a single-objective technique and can be clubbed with NSGA-II, a multi-objective technique, and this combination seems to be a powerful tool to handle very large-scale problem. Ponnambalam et al. (2007) proposed a GA based technique in their work using non binary real valued workload data as an input matrix and developed a modified grouping efficiency. Their method seemed to outperform traditional techniques such as K-mean clustering and ART1 algorithms. Pillai and Subbarao (2007) designed GA as robust design methodology which works with a forecast of product mix and demand changes from period to period of a planning horizon and does not allow the composition of machine cells to change over time. James et al. (2007) demonstrated a hybrid GGA technique combined with local search for CFP which reduces variability of the solutions obtained and outperforms many well-known techniques including conventional GGA. Tavakkoli-Moghaddam et al. (2007) assumed demand of parts to be dynamic and uncertain in fuzzy environment and developed an integer coded GA method to handle any size of the given problem. Boulif and Atif (2008) considered dynamic production factors like input data, with realistic constraints and avoiding assumptions like static number of cells, hence proposed a better GA based methodology with the help of fuzzy logic. Mahapatra and Pandian (2008) studied the operational time and sequence of operation of parts, to minimize cell load variation and exceptional elements by applying GA methods. The solution outperforms K-mean clustering technique and C-link clustering algorithms. Chan et al. (2008) introduced CFP with IAECLP with two objectives of minimizing intracell and intercell part movement and total sum of intracell and intercell part distance unit due to machine sequence and sequences of newly formed cells and then applied GA on top of it for better result. Kao et al. (2008) presented a new DE-based algorithm to solve cell formation problems, namely DECF algorithm. Each chromosome vector represents a solution which contains machine and part cluster centers together. A set of chromosome vectors iteratively moves to a better position in a continuous search space through three operations of mutation, crossover and selection. The experimental results show that DECF can compete with other well established methods. Defersha and Chen (2008a) developed a

mathematical programming model integrated with cell configuration and lot sizing in a dynamic manufacturing environment and implemented a hybrid GA embedded with linear programming technique, and reported that a simplex method can be used to solve the linear programming sub-problem which in turn can generate near optimal solution efficiently. Defersha and Chen (2008b) further used parallel GA with island model for dynamic cell formation problem with parameters including connection topology, migration policy, migration frequency migration rate, and a repair heuristic. The authors demonstrated that the model could outperform previous sequential methods. Tariq et al. (2009) developed a local search heuristic based GA as a methodology of CFP, which uses integer type representation, multi-point crossover and roulette wheel selection procedure which yields best solution ever found in literature. Tunnukij and Hicks (2009) presented the enhanced grouping genetic algorithm (EnGGA) to solve the CFP without predetermining the number of manufacturing cells or the number of machines and parts within each cell. The method replaces the replacement heuristic in a standard GGA with a greedy heuristic and employs a rank-based roulette–elitist strategy, as a new strategy for creating successive generations. Output of EnGGA outperforms other traditional methods. Another study shown (Mahdavi et al., 2009) that cell formation with an objective of minimizing total number of voids and EEs in part-machine incident matrix by using a GA embedded with a heuristic inspired mutation is efficient and it yields significantly improved solution. Haleh et al. (2009) developed new hybrid technique based on memetic algorithm and revised TOPSIS method called (HMA-RTM) and applied on multi-objective CFP based on total cell moves and cell load variation and compared the result with GP-SLCA method, satisfactory output obtained. Cao et al. (2009) formulated a mathematical model for optimal lot splitting into alternative routes to account for either positive or negative effects of production run length on product quality in a cellular manufacturing environment. Optimal lot splitting is required to balance the cost of inter-cell material handling and the cost of replacing defective parts. They also developed a heuristic method based on a genetic algorithm for the proposed model for large-scale problems, and the solutions found by the developed heuristic method were very encouraging. Kor et al. (2009) aimed to implement SPEA-II method for multi-objective CFP and compared with GP-SLCA method, which produced good result. Bajestani et al. (2009) presented a new multi objective scatter search (MOSS) for dynamic CFP with two objectives of minimizing total cell load variation and sum of the miscellaneous costs. A memetic algorithm also introduced for the best next-population solutions to generate diverse initial solution and the results indicated superiority over SPEA-II and NSGA-II. The methodology proposed by Noktehdan et al. (2010) introduced a differential evolution (DE) approach by combining the features of grouping genetic algorithm (GGA) to solve CF problems and compared the optimality of solutions effectively with previous research data and found better grouping efficacy. Fan et al. (2010) discussed the dual resource-constrained system model for CFP, where the minimum distance of parts and employees move among cells, the number of hired employees and the load balance of staff are all considered and a GA was used to solve simple numerical example to validate the model. Pailla et al. (2010) proposed two methodologies for CFP, one was a modified evolutionary algorithm based on genetic operator-heuristic and the other was based on simulated annealing. The experimental result indicated that the evolutionary technique was an efficient local search mechanism which could reduce the CPU time in terms of the number of iterations and the SA method could outperform every technique including the former evolutionary methodology. Neto and Filho (2010) designed a multi-objective optimization model using GA for CFP, where fitness evaluation was performed via simulation of cellular system where congestion effect was incorporated and dynamic routing policy was used. Computational result exhibits improvement in terms of WIP level, intercell movements by reducing machine investment. The work proposed by Deljoo et al. (2010) based on dynamic production condition considered as factors affecting CF problems such as, product mix, demand of parts during some period, machine movement, addition of new equipment, providing flexibility in cellular manufacturing, which was further solved using some modified GA.

The abovementioned EA based literature survey criticises several infinitesimal issues related to the techniques proposed by researchers. Tables 4a to 4d render the detailed outcomes of individual methodologies.

Table 4a
Various attributes of the proposed EA based methodologies

References	Initial Population	Fitness function	Selection strategy	Stopping Criteria
Venugopal and Narendran (1992)	randomly generate the initial population	Total intercell moves and within cell load variation	*stochastic remainder selection* without *replacement* scheme	Fixed no. of iteration
Gupta et al. (1996)	Randomly generate the initial population	Objective function taken	*stochastic remainder selection* without *replacement* scheme	Fixed no. of iteration
Joines et al. (1996)	Random seeding	Nonlinear form of grouping efficacy	Normalized geometric ranking scheme	maximum number of generations
Morad and Zalzala (1996)	initial population is generated at random	Objective function taken	elitist strategy	maximum number of generations
Hwang and Sun (1996)	permutations generated with the numbers	Scaled fitness $sf_{ji} = fitness + offset /(sum (fitness/PS + offset)$	stochastic remainder sampling without replacement	maximum number of generations
Zhao et al. (1996)	randomly generated by heuristic	rank - based evaluation function	roulette wheel approach	maximum number of generations
Kazerooni et al. (1997)	randomly generated	number of elements in the MCS matrix which have a value equal to zero or below L_n	tournament strategy	maximum number of generations
Suer (1997)	randomly generated	total number machine types	reproduction probability based scheme	Fixed no. of iteration
Al-Sultan and Fedjki (1997)	random generation	objective function value	biased roulette wheel approach	maximum number of generations
Pierreval and Plaquin (1998)	randomly generating algorithm	total cost or the homogeneity of the workload distribution on each cells	niched pareto tournament selection	If all the machines are placed in cell
Gravel et al. (1998)	generated randomly	objective function value	chosen by fitness	When the diversity drops to zero or loss of diversity of the machine cell population should not exceed 3%.
Hsu and Su (1998)	generated randomly	total cost, and total machine loading imbalances	chosen by fitness	maximum number of generations
Moon and Gen (1999)	generated randomly	objective function value	Deterministic selection strategy	maximum number of generations
Zhao and Wu (2000)	generated randomly	objective function value	chosen by fitness	maximum number of generations
Mak and Wong (2000)	Generate an initial population of individuals randomly	objective function values	chosen by fitness	maximum number of generations
Mak et al. (2000)	Randomly generated	Bond energy measure	traditional roulette wheel selection operator	maximum number of generations
Lee-Post (2000)	Generate randomly	sum of similarities	selected probabilistically	time-bounded rule & quality-bounded rule

Table 4b

Various attributes of the proposed EA based methodologies

References	Initial Population	Fitness function	Selection strategy	Stopping Criteria
Plaquin and Pierreval (2000)	generated randomly	inter-cell traffic function	Based on aggregates and their belongingness	When there is no aggregate left to place
Onwubolu and Mutingi (2001)	randomly created solution space	Cost function	remainder stochastic sampling without replacement	maximum number of generations
Chu and Tsai (2001)	variable restriction method **to** generate randomly	minimizing the number of EEs	roulette wheel selection method	number of generations
Dimopoulos and Mort (2001)	randomly created similarity coefficients	Grouping efficacy and weighted grouping efficiency	Tournament selection	number of generations
Brown and Sumichrast (2001)	Random generation	Based on objectives	rank-based roulette-wheel selection	number of generations
Chi and Lin (2002)	initial radius of the hyperboxes	Objectives and grouping efficiency	stochastic sampling method without replacement	Fixed no. of iteration
Wu et al. (2002)	randomly generate the initial population	Total number of EEs	roulette wheel approach	maximum number of generations
Zolfagharia and Liang (2003)	randomly generated	generalized grouping efficacy	random selection, roulette wheel selection, stochastic universal sampling	maximum number of generations
Mansouri et al. (2003)	Randomly Generate Initial Solutions	$Fi = Ci/(Ci + fi)$ Ci=normalize factor, fi=objective value	*Reminder Stochastic Sampling Without Replacement* in conjunction with a new Elitism operator	either it converges to a robust non-dominated frontier or a predetermined number of generations
Chan et al. (2004)	random population	$\Gamma a = Zbest/_{Za}$ Za = objective value of the alternative	Individuals with higher fitness value	variation in the value of the best objective function
Chi and Yan (2004)	generated randomly	Fuzzy objective function	roulette wheel approach	maximum number of generations
Goncalves and Resende (2004)	randomly generated	objective function	elitist strategy	Maximum No. of generation
Solimanpur et al. (2004)	randomly generated	Total objective function	Probabilistic selection	Maximum No. of generation
Zolfaghari and Liang (2004)	randomly generated	Based on objectives	Best fit parents selected randomly	Maximum No. of generation
Muruganandam et al. (2005)	Generated randomly	$F(t) = {}^{1}/_{1 + f(t)}$ f(t) = objective value	roulette wheel selection method is adopted	maximum number of generations
Pai et al. (2005)	generated randomly	grouping efficacy	roulette wheel selection principle	maximum number of generations
Vin et al. (2005)	Generate an initial population using a resource planning (RP) heuristic	Cost function	Individuals with higher fitness value	maximum number of generation without improvement

Table 4c

Various attributes of the proposed EA based methodologies

References	Initial Population	Fitness function	Selection strategy	Stopping Criteria
Rogers and Kulkarni (2005)	randomly generated	objective function + penalty function	standard proportional selection incorporating the elitist model	Maximum No. of generation
Rajagopalan and Fonseca (2005)	randomly generated	Production volume function considering upper limit and lower limit of VSM	tournament selection	Maximum No. of generation considering upper limit and lower limit of VSM
Hu and Yasuda (2005)	Random heuristic	$Fitness = -A1 \times C \times f1 - A2/c \times f2$	probabilistic selection	Maximum No. of generation
Rajagopalan and Fonseca (2006)	randomly generated	material handling cost + penalty cost	tournament selection	a run of 5000 generation
Filho and Tiberti (2006)	special procedure based on random generation	Sum of the objectives	Roulette Wheel selection procedure	Maximum No. of generation
Nsakanda et al. (2006)	randomly generated using population diversity	Total move cost + total outsourcing cost	stochastic remainder selection without replacement method	No. of generation, number of chromosomes evaluations exceeds, improvement in fitness value, population diversity drops
Boulif and Atif (2006)	randomly generated initial population	objective function	Roulette wheel random procedure	Maximum No. of generation
Chan et al. (2006)	initially generated randomly	$\Gamma a = Zbest/Za$ Za = objective value of the alternative	Chromosomes with higher fitness value	little change of improvement in the best objective function
Defersha and Chen (2006)	Random generation	Sum of the objectives	biased roulette wheel approach where each individual chromosome in the current population has a roulette wheel slot sized in proportion to its transformed fitness	Maximum No. of generation
Wu et al. (2006)	randomly generated	$f = \sum_{i}^{popsize} [g(i)/g(x)]$ g(x) = objective function	roulette wheel and elitist approach	Maximum No. of generation
Car and Mikac (2006)	random selection of individuals	sum of total number of voids and the total number of EEs	Individuals with higher fitness value	Maximum No. of generation
Dimopoulos (2006)	randomly created similarity coefficients	objective value	tournament selection	Maximum No. of generation
Ponnambalam et al. (2007)	generated randomly	objective function	maximum fitness function value	Maximum No. of generation
Pillai and Subbarao (2007)	randomly created population	objective function	Best fit chromosomes	Maximum No. of generation
James et al. (2007)	Random generation	$f = 2r/N(N+1)$ r = rank; N = no. of ranked chromosomes	Rank-based roulette wheel selection	No. of generation
Tavakkoli-Moghaddam et al. (2007)	greedy generational handling strategy	objective function + penalty function	roulette wheel sampling	Maximum CPU time, standard deviation of generation,

Table 4d
Various attributes of the proposed EA based methodologies

References	Initial Population	Fitness function	Selection strategy	Stopping Criteria
Boulif and Atif (2008)	Random generation	objective function	roulette wheel approach	Maximum No. of generation
Chan et al. (2008)	random population	objective function	Best fit chromosomes	little change in the best objective function
Kao et al. (2008)	Random generation	Grouping Efficacy	Best fit chromosomes	Fixed no. of iteration
Defersha & Chen (2008a)	Random generation	Sum of the objectives	biased roulette wheel approachis simulated as suggested by Goldberg (1989)	No. of generation, improvement in fitness value
Defersha & Chen (2008b)	Random generation	Sum of the objectives	biased roulette wheel with replacement	improvement in fitness value
Mahapatra & Pandian (2008)	Generate random population	objective function	Random selection	Maximum No. of generations
Haleh et al. (2009)	Generate random population	objective functions and RTM method used	Best fit chromosomes	Maximum No. of generations
Mahdavi et al. (2009)	special procedure was developed	total number of voids and EEs	Roulette Wheel selection procedure	Maximum No. of generations
Tariq et al. (2009)	Random generation	objective function	Best fit chromosomes & roulette wheel approach	improvement in fitness value
Tunnukij and Hicks (2009)	Random generation	Grouping efficacy	Random selection & Rank-based Roulette-elitist strategy	Maximum Number of generation
Kor et al. (2009)	Random generation	closeness to the true Pareto front and even distribution of solutions	Binary tournament selection with replacement	Maximum No. of generations
Cao et al. (2009)	Random generation	Objective function value	Best fit chromosomes	No. of generation, improvement in fitness value
Bajestani et al. (2009)	Based on memetic procedure	Objective function value	Similarity-rate function is used	Maximum No. of generation
Neto & Filho (2010)	first half is generated by using problem-specific information & second half is generated randomly	Feasibility correction is used to check objective value therefore fitness	NSGA-2 built-in "crowding" tournament used	Maximum No. of generation
Pailla et al. (2010)	Random Generation & constructive heuristic used	grouping efficacy	Selection probability function used from Joins et al. (1996)	Maximum No. of generation
Noktehdan et al. (2010)	truncated geometric distribution	Cost Function	Best fit chromosomes	Maximum No. of generation
Fan et al. (2010)	Random generation	Objective function of CFP used	Roulette wheel method	Maximum No. of generation
Deljoo et al. (2010)	Sequential strategy used	Objective function of CFP used	Best fit chromosomes taken & normalized method used	No. of generation, upper bound of solving time, improvement in fitness value

2.2.6 Ant Colony Optimization (ACO)

The first ACO algorithm appeared in early 90s by Dorigo and ACO is now a widely studied meta-heuristic for combinatorial optimization problems, as the recent book by Dorigo and Stutzle (2004) testifies. The concept is based on the observation of foraging behaviour of ants: when walking on routes from the nest to a source of food, ants seem to find not just a simple random route, but a quite 'good' one, in terms of shortness, or equivalently, in terms of time of travel; thus, their behaviour allows them to solve an optimization problem. This kind of accomplishment of biological ants can be explained by the way of communication and choosing the right way to go. In fact, when an ant starts walking, it normally deposits a chemical called pheromone on the ground and the pheromone usually disappears during the time since it evaporates. Therefore, the more pheromone exists in a particular place, the more chance to have food around and this could help ants find shorter routes to choose food since there are more pheromone on these routes. This is basic idea of ACO algorithm. Following pseudocode 4 shows ACO approach,

2.2.7 ACO in Cell Formation

ACO is successfully implemented on CFP with various flavours of the algorithm. Islier (2005) proposed artificial ant system to optimize CF problems by means of touring of Ants from minor places (machine/component) to major places (cells). The methodology was compared with previous AI techniques such as SA, TS, GA and shown GA is only comparable with the ant system methodology. Prabhaharan et al. (2005) also proposed an ant colony system (ACS) approach for CFP to minimize total cell load variation and intercell moves considering demands for number of parts, routing sequences, processing time, machine capacities, and machine workload status. The result shown ACS outperformed existing GA method in several cases. Giri et al. (2007) reported a TSP based heuristic embedded with ACO to form optimal part/machine clusters based on the overall machine sequence which leads to configurations with a minimum number of movements among the cells. The ACO is used here to obtain optimum machine sequence for maximum part volume flow. Since TSP was used hence graph theory based approach was also followed. Mak et al. (2007) studied the virtual cellular manufacturing system (VCMS) and explained that the application of an ACO with minor modification and adding some heuristics to produce efficient manufacturing cells could reduce the cost of production schedule by minimizing material handling cost. The methodology was applied for some combustion engine manufacturing company and the result was compared with GA where the proposed technique performs better in terms of reduced computational time. Kao et al. (2008) proposed ant colony clustering (ACC) technique to solve CF problems. Their focus area was to form efficient part families by the chemical recognition phenomena followed by ants to form cluster. They also showed improved solution and found optimal grouping efficiency. Megala et al. (2008) suggested a modified ant colony optimization algorithm based on ACO to solve CFP with existing data sets from literature and the results showed the ability of the algorithm to maximize grouping efficacy. Spiliopoulos and Sofianopoulou (2008) developed new ACO based robust methodology, used tight eigen-value based bound to differentiate solutions to accelerate the search. The method is applied on cell design problem with maximum cell size and the processing sequence of the parts, and avoids difficulties associated with the use of the traditionally used part-machine incidence matrix and improved result found for medium-to- large size problems. Solimanpur et al. (2010) proposed ACO to solve CFP with the consideration of operational sequence and production volume with the objectives to reduce intercell moves and number of voids, and the ACO converges to optimality. Li et al. (2010) proposed MAX-MIN ant system integrated local search technique for ACO-CF model implemented in hyper-cube framework. The result is not only better than previous techniques, it can also increase efficacy by allowing residual cells in diagonal blocks. Xing et al. (2010) proposed two part-machine clustering techniques, one is with ART1 neural network based approach and another is ant colony

system (ACS) based approach, the computational result found is ACS is better than the ART1 method, by using grouping efficiency measure.

Pseudocode 4: Ant Colony Optimization (ACO)

Initialize;
i← 0;
repeat
        generate a feasible solution;
        evaluate goodness η;
        if $\eta > \eta max$ then begin update elite list; shift the bounds end;
        if i mod $\alpha = 0$ then alter the solution;
        if i mod $\beta = 0$ then intensify elite pheromone traces;
        update pheromone trails;
        i← i + 1;
until i=σ;

2.2.8 Particle Swarm Optimization (PSO)

PSO algorithm was first proposed by Kennedy and Eberhart (1995) in the mid-90s, which is one of the latest evolutionary optimization techniques. PSO is inspired by the metaphor of social interaction and communication in a flock of birds or school of fishes. In these groups, there is a main agent who guides the movement of the whole swarm. The movement of every individual is based on the main agent and on his own knowledge. PSO is population-based and evolutionary in nature. Therefore, particles in a PSO method normally follow the main agent which is the one with the best performance. The pseudocode 5 suggests the steps of PSO. It can be observed that PSO is more efficient and less complex than other population based method applied in CF domain.

2.2.9 PSO in Cell Formation

Andres and Lozano (2006) developed very first PSO technique for cell formation problem. The solution encoding corresponds to a vector of particle-position, which are to be updated with the iterations. The approach indicated that PSO has a greater capability of finding optimal solution in reasonable amount of time. Ming and Ponnambalam (2008) proposed a hybrid PSO approach combined with GA for CF problem and PSO to find optimal layout, the methodology considered randomly generated initial particles and velocities and it was successfully applied to minimize total cell load variation and total components move. Durán et al. (2010) reported a modified PSO with proportional likelihood instead of using velocity vector on CF problems where the objectives are the minimization of cell load variation and inter cellular parts movement and reported the stability of the method with low variability. Mehdizadeh and Tavakkoli-Moghaddam (2009) proposed a Fuzzy PSO (FPSO) technique to solve CF problem in the context of part-machine clustering where each particle corresponds the cluster center vector and swarm represents a number of candidates clustering for the current data vector and they showed that for a large-scale problem the proposed technique could produce better solution. Caprihan et al. (2009) stated a quantum PSO (QPSO) method and designed a virtual cellular manufacturing system (VCM) and the proposed method was tested with GA and lexico goal programming approach where QPSO approach consumed less CPU time and yielded better solution. A similar study was also performed by Anvari et al. (2010) where a hybrid particle swarm optimization technique for CFP was reported. The initial solutions generated either randomly

or using a diversification generation method and the technique also utilized mutation operator embedded in velocity update equation to avoid reaching local optimal solutions. Thereafter with due consideration, a wide variety of machine/part matrices were effectively solved by this approach.

Pseudocode 5: Particle Swarm Optimization (PSO)

Initialize;
repeat
  Evaluate fitness for each particle;
  Update the global best and local best position.
  Update particle velocity by v[i+1] = w0v[i]+c1*rand()*(pbest[i]
  present[i])+c2*rand()*(gbest[i]-present[i])
  Update particle position by present[i+1] = present[i]+v[i]
Until maximum number of generation reached

2.2.10 Bees Algorithm (BA)

One of the newest techniques evolved in this genre is Bees algorithm invented by D.T. Pham (2006). The BA is an optimization algorithm inspired by the natural foraging behaviour of honey bees to find the optimal solution. The phenomenon behind this algorithm is the food foraging behaviour of honey bees. Honey bees are normally able to extend their colony over long distances and in various possible directions simultaneously to take advantage of substantial number of food sources. A colony succeeds by redistributing its foragers to suitable fields. Normally, more bees must be recruited for flower patches with ample amounts of nectar or pollen that can be gathered with less effort. The pseudocode 6 demonstrates the BA procedure.

2.2.11 BA in Cell Formation

BA is successfully implemented in CF domain by Pham et al. (2007) in order to reduce intracell and intercell moves by considering bond energy and grouping efficacy measure. The initial solutions generated randomly with certain number of scout bees. In the searching phase more scout bees are assigned in the vicinity of best sites which are selected according to computed fitness values. The algorithm shows its highly competitive nature to obtain optimal solution when compared with other established methods.

Pseudocode 6: Bees Algorithm (BA)

Initialize;
repeat
  Evaluate fitness of the population.
  while (stopping criterion did not meet)
    Select sites for neighbourhood search.
    Recruit bees for selected sites (more bees for the best *e* sites).
    Evaluate fitnesses.
    Select the fittest bee from each site.
    Assign remaining bees to search randomly and evaluate their fitnesses.
  end while
until maximum number of generation reached

2.2.12 Water Flow-like Algorithm (WFA)

WFA was first proposed by Yang and Wang (2007) as a nature inspired optimization algorithm for object clustering, to overcome the shortcoming the single and multiple-solution-agent-based algorithms. It mimics the behavior of water flowing from higher to lower level which helps in the process of searching for optimal solution. WFA is given in pseudocode 7.

2.2.13 WFA in Cell Formation

Wu et al. (2010) introduced water flow-like algorithm (WFA) in CFP, which deals with dynamic size of solution agents, overcomes the drawbacks of single agent based and multi agent based techniques. WFACF model proposed by the researchers utilizes similarity coefficients method and machine assignments and part assignments method to generate initial solution for later stage and flow splitting and moving operation are employed to obtain better neighbourhood solutions. The method has two stages; the first step produced feasible solutions without substantial improvement in solution to derive a cell size quickly, which is then implemented as input to the second stage to detect the near-optimal solution. The result shown is better than existing procedures.

Pseudocode 7: Water Flow-like Algorithm (WFA)

Initialize;
repeat
    repeat
        calculate no. of subflows
        for each subflow find best neighbourhood solution
        distribute mass of flow to its subflows
        calculate improvement in objective value
    until population no. reached
    merge subflows with same objective values
    update the no. of subflows
    update total no. of water flows
    if precipitation condition met
        perform bit reordering strategy
        distribute mass to flows
        evaluate new solution
        update the no. of subflows
        update total no. of water flows
Until maximum generation reached

---

3. Discussion

This section takes a transversal view on the reviewed meta-heuristics and points out some open issues and possible direction of future study.

3.1 Comparison based on objective function

CF problems can be formulated using single objective or multiple objectives, such as intercell or intracell part movement, within cell load variation, count of EEs and voids, machine utilization, machine investment, machine duplicacy, WIP level etc. by considering operational time, operational sequence of parts. Table 5 classifies literatures studied based on multi-objectives with production

factor considered. Around 80% of the papers listed in Table 2 are bi-objectives, and around 50% amongst them comprised total cell movements and cell load variations.

Table 5
List of papers with multi-objective CFPs

References	Obj1	Obj2	Obj3	Obj4	Obj5	Obj6	Obj7	Obj8	Obj9
Neto & Filho (2010)	✓	✓	✓						
Zhao & Wu (2000)		✓		✓	✓				
Brown & Sumichrast (2001)		✓				✓			
Gupta et al. (1996)		✓		✓					
Hsu & Su (1998)		✓	✓	✓					
Mansouri et al. (2003)		✓					✓	✓	
Solimanpur et al. (2004)			✓						✓
Yasuda et al. (2005)		✓		✓					
Wu et al. (2006)		✓			✓				
Dimopoulos (2006)		✓				✓			
Tavakkoli-Moghaddam et al. (2007)		✓	✓						
Defersha & Chen et al. (2008)		✓	✓						
Goncalves & Resende (2004)		✓				✓			
Gravel et al. (1998)		✓		✓					
Chi & Yan (2004)		✓		✓					
Fan et al. (2010)		✓		✓					
Morad & Zalzala (1996)		✓		✓					
Kor et al. (2009)		✓		✓					
Mahapatra & Pandian (2008)				✓	✓				
Mak & Wong (2000)		✓		✓					
James et al. (2007)		✓				✓			
Haleh et al. (2009)		✓		✓					
Tariq et al. (2009)		✓				✓			
Muruganandam et al. (2005)		✓		✓					
Bajestani et al. (2009)		✓	✓						
Li et al. (2010)		✓				✓			
Solimanpur et al. (2010)	✓	✓						✓	
Prabhaharan et al. (2005)		✓		✓					
Ming & Ponnambalam (2008)		✓		✓					
Su & Hsu (1998)		✓		✓					
Das et al. (2006)		✓						✓	
Mahesh & Srinivasan (2006)				✓		✓			✓
Lei & Wu (2006)		✓		✓					
Jayaswal & Adil (2004)		✓	✓						
Vakharia & Chang (1997)		✓	✓					✓	
Foulds et al. (2006)		✓	✓						
Tavakkoli-Moghaddam et al. (2005)		✓	✓						

Obj1: Level of WIP
Obj2: intercell and/or intracell move
Obj3: Machine investment/modification/relocation
Obj4: Cell load variation
Obj5: Count of EEs and/or Voids/Operational sequence/time
Obj6: machine utilization/cycle time of parts
Obj7: machine duplication & part subcontracting
Obj8: system under-utilization/ cells utilization/system reliability
Obj9: part processing time/cost/total work content of parts

3.2 Comparison among different meta-heuristics

Most of the papers in the CFP literature focus on single meta-heuristic approach, which is compared either to the variants of the same technique, or to previously available methods such as similarity coefficient method, mathematical programming method, or to simple heuristics such as random search, greedy search, or to exact methods when these are available. Few papers perform comparisons among different meta-heuristics. Table 6 summarizes various meta-heuristic methods used and compared for CF problems.

Table 6
Papers with comparison between meta-heuristics performance

References	Meta-heuristics compared	Winner	Tool used	Winning %
Vakharia and Chang (1997)	SA, TS	SA	**	
Tavakkoli-Moghaddam et al. (2005)	TS, SA, GA	SA	VB 6	
Noktehdan et al. (2010)	GDE, GGA	GDE	Matlab 7.4	
Pailla et al. (2010)	EA, SA, HGA	SA	**	
Wu et al. (2009)	HSAM, SA, TS,	HSAM	C	
Wu et al. (2010)	WFA, SA, HGA	WFA	C	**WFA 4%**
Attila Islier (2005)	Ant System, TS, SA, GA	Ant System	**	
Mak et al. (2007)	GA, ACO	ACO	VC++ .NET	
Goncalves & Resende (2004)	EA, GA, GP	EA	VO 2.0b-1	
Mahdavi et al. (2009)	GA, SA, EA	GA	Matlab 7	
Li et al. (2010)	ACO, EAs	ACO	C	
Solimanpur et al. (2010)	ACO, GA	ACO	C	
Spiliopoulos & Sofianopoulou (2008)	ACO, TS	ACO	Fortran 90	
Prabhaharan et al. (2005)	ACO, GA	ACO	**	**ACO 21%**
Durán et al. (2010)	PSO, SA	PSO	**	
Caprihan et al. (2009)	QPSO, GA	QPSO	**	**PSO 7%**
Safaei et al. (2008)	MFA, SA, MFA-SA	MFA-SA	**	
Lei & Wu (2006)	MOTS, GA, PSA	MOTS	**	
Arkat et al. (2007)	SA, GA	SA	**	
Wu et al. (2009)	SA, GA	SA	C	**SA 25%**
Onwubolu & Songore (2000)	TS, SA	TS	PASCAL 7.0	
Adenso-Diaz et al. (2001)	SA, TS	TS	**	**TS 11%**
Bajestani et al. (2009)	MOSS, SPEA, NSGA	MOSS	Matlab 7.0	
Muruganandam et al. (2005)	MA, GA, TS	MA	C	
Haleh et al. (2009)	MA, GP	MA	**	
James et al. (2007)	GP, EA, HGGA, GA	HGGA	VB .NET	
Tunnukij & Hicks (2009)	GA, SA, TS, EnGGA	EnGGA	C	
Yasuda et al. (2005)	SA, GGA	GGA	Matlab 6.0	**EA 32%**

**: Data not available

The table also depicts the winning alternative as well as the percentage of the success. As it can be observed, EA has the highest rate of success with 32% amidst other meta-heuristics such as SA, ACO, TS, PSO, and WFA. Therefore, it is understandable that frequency of usage and winning capability are higher for EA than the other meta-heuristics. However, there are chances for other meta-heuristics

to be used and compared with the previous works under different situations and production conditions. Thus, it can be concluded that it is still impossible to give guidelines on which meta-heuristic is better in which situation. One important aspect is that experimentation with different meta-heuristic approaches could reveal, whether the effectiveness of a meta-heuristic is due to the particular adjustments to speed up the computation such as approximating the objective function or using carefully designed performance measures and statistical comparison strategies or proper adjustment in parameters. The pie chart shown in Fig. 2 presents the winning rate of the meta-heuristic techniques for CF problems derived from Table 6.

3.3 Use of hybrid techniques or combination of two or more meta-heuristic techniques

There are only finite numbers of research works focused on hybrid techniques utterly based on various meta-heuristics in CFP area. This is also an emerging research area where new techniques based on meta-heuristics can be formed. Table 7 shows the number of papers available in the field of CFP which demonstrate the possibility, the effectiveness and the usability of such hybrid meta-heuristic techniques. The two different forms of hybridization are demonstrated from literature (Bianchi et al., 2009), such as, (1) component exchange among meta-heuristics (CEM), (2) cooperative search (CS).

- CEM: It performs inclusion of component of one meta-heuristic into another. The basic idea behind this is that population-based methods are efficient in identifying promising area in search space and deterministic or single solution based methods are good in exploring the promising area. Hence hybridisation is required. It can also be stated that a solution obtained from recombination operation of population based method (GA) is usually more different from the parent than a predecessor solution to a successor solution obtained by applying a move in deterministic method (TS).

- CS: It brings parallelism in meta-heuristics execution with different level of communication. It is possible for different meta-heuristics to exchange information about states, models, solutions, sub-problems, different attributes of search space. Hence combining them as cooperative search technique can enhance the performance and combine characteristics and strengths of them.
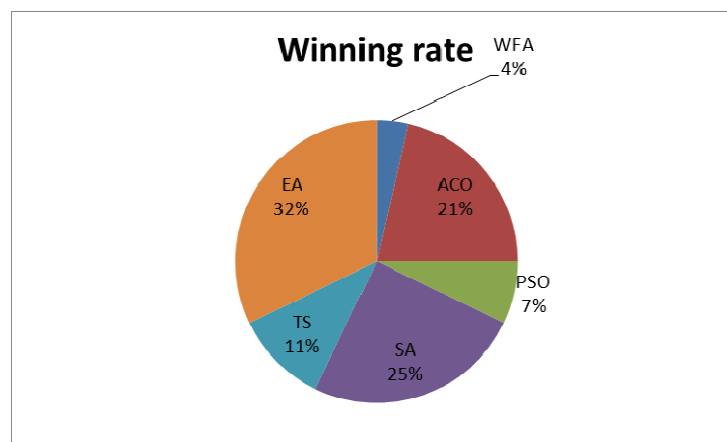

Fig. 2. percentage (%) of winning of each meta-heuristic

Table 7
Papers based on hybrid meta-heuristic method

References	Problem model	Meta-heuristics
Wu et al. (2009)	NLP	TS-mutation
Ming & Ponnambalam (2008)	QAP	GA-PSO
Ateme-Nguema & Dao (2007)	BIP	ACO, TS
Su & Hsu (1998)	MOMP	SA, GA
Das et al. (2006)	IP	SA,GA
Wu et al. (2007)	NLP	SA, GA
Safaei et al. (2008)	MIP	MFA, SA

3.4 Usage statistics of meta-heuristics in cell formation problems

Fig. 3 demonstrates a Pareto-analysis of the frequency of usage of the meta-heuristics and also the cumulative usage of these techniques in CF domain where EA is winner in most cases and SA and TS are believed to be effective methods as well. Cumulative usage indicates that nearly 29% of the techniques (EA and SA) accounted for around 73% of usage.

Finally Fig. 3 depicts that other meta-heuristics which are strongly competitive with EA, SA, TS and ACO, such as PSO, BA, WFA, are used fewer number of times.

Almost all the recently developed meta-heuristics are used to solve single objective CFPs, and only few are developed to handle multi-objective problems. Hence, the combined conclusion drawn from Fig. 2 and Fig. 3 is that the usage of EA and winning possibility of EA is higher according to the research work done till now in CF domain. While comparing with recent review work proposed by Papaioannou and Wilson (2010), this present work introduced more intricate study in cellular manufacturing. The uniqueness of this paper is to put major concentration in meta-heuristics based approaches and a detailed discussion based on many critical issues as stated above.

From the study presented in this paper, the followings are concluded,

a. EAs are proven techniques in engineering optimization problem, reflection is found in CF domain as well. Since late 90s GA is proposed by many researchers as a stand-alone tool and also as a hybrid technique and being used rigorously till present time in search of better solutions.

b. In early stages single objective CFP was of researchers' prime interest, but in later stage since manufacturing decisions are becoming more complex, so multi-objective CFPs are taken up by considering operational time, sequence, alternative process routing, machine duplicacy, dynamic conditions, and various costs related to CMS.

c. Considering the fact that multi-objectivity is difficult to deal with in CFP, and to tackle these problems several multi-objective EA methods are appropriate such as NSGA, SPEA, NPEA and MOGA. Therefore, the ones which are frequently used by researchers, reported in Table 5, such as PSO, ACO, BA, WFA are still in developing stage we may also solve CFP with multi-objective algorithms of PSO, ACO, BA, WFA as well.

d. For many large-scale problems, computational time is a major concern of many researchers, and hence better evolutionary optimization techniques are being proposed accordingly.

e. From late 90s GA, TS, SA are mostly considered techniques in CF domain as an optimization tool.

f. New population based tools such as PSO, ACO are attracting more research interests since they are computationally more attractive and less complex.

g. Many other tools such as BA, WFA, scatter search and other hybrid techniques are also evolving with time as CF solution methodologies.

h. In case of hybridization, *component exchange among metaheuristics* is used but *cooperative search* is yet to be fully utilized.

i. In case of comparing two or more metaheuristics in a single research work, EA is used mostly and winning rate over other algorithms is (32%) from Fig. 2, and till now usage of EA is (52.67%) which is the highest. Thus EA seems to be the strongest while its usage proceeding towards certain saturation level.

j. Although SA stood the second in using (19.84%) and winning (25%), but result does not tally with TS and ACO. TS is older technique, usage rate is the third highest (13.74%) but in case of winning among other metaheuristics, it (10.71%) comes after ACO (21.42%). Spiliopoulos and Sofianopoulou (2008) shown ACO could outperform TS in solving CFP, which is a clear indication why ACO is the third winner in Fig. 2.
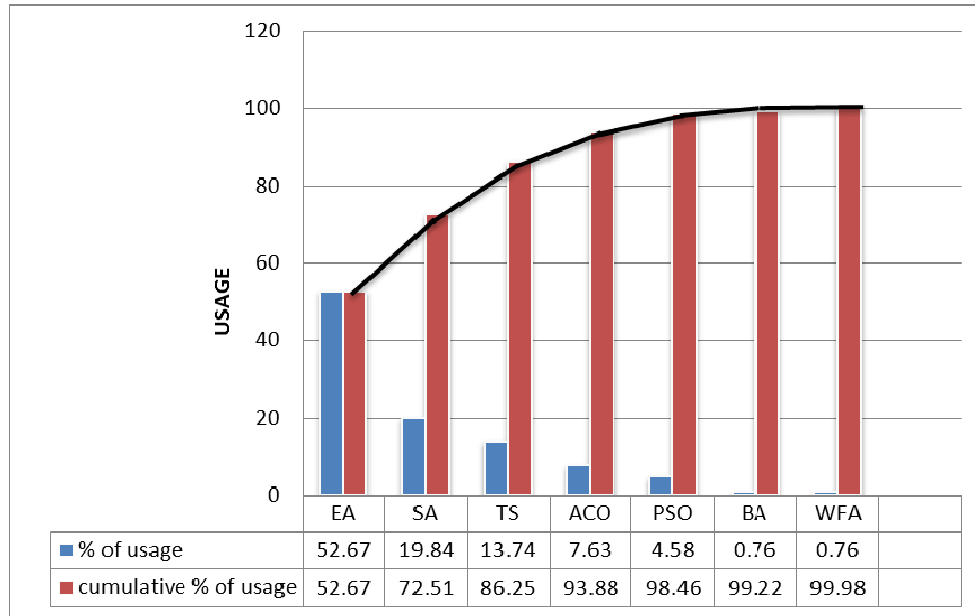

	EA	SA	TS	ACO	PSO	BA	WFA
■ % of usage	52.67	19.84	13.74	7.63	4.58	0.76	0.76
■ cumulative % of usage	52.67	72.51	86.25	93.88	98.46	99.22	99.98

Fig. 3. Usage statistics of meta-heuristics

4. Conclusion

This paper presents an in depth review of recent CF based meta-heuristic methodologies. Since mid-90s EAs, SA, TS have evolved as powerful optimization techniques in CFP and a substantial volume of research papers are available which are focused on these techniques. Amongst these, GA has been the most adaptable technique for the researchers when it is combined with other algorithms and dominating as a solution methodology in Cellular Manufacturing since the last two decades. ACO, PSO, BA and WFA are lately developed and able to compete with GA. Therefore, it seems that the trend is on using population based methodologies as well. According to our survey, it seems that the new methodologies are built combined with GA to solve CFP as hybrid techniques. As a future research, it is possible to use hybrid techniques to solve more realistic and complex GT/CM problems. For example, ACO-PSO, GA-PSO, BA-PSO, WFA-GA, SA-GA, TS-GA and other similar approaches would be perfect methods to handle large-scale industrial optimization problems in aforesaid domain with due focus on higher efficiency due focus on less computational time and higher efficiency.

References

Aarts, E. & Korst, J. (1990). *Simulated Annealing and the Boltzmann Machine*. John Wiley & Sons, New York, USA.

Abduelmola, A. I. & Taboun, S.M. (2000). A simulated annealing algorithm for designing cellular manufacturing systems with productivity consideration. *Production Planning & Control,* 11 (6), 589-597.

Adenso-Diaz, B., Lozano, S., Racero, J. & Guerrero, F. (2001). Machine cell formation in generalized group technology. *Computers and Industrial Engineering*, 41, 227-240.

Adil, G. K. & Rajamani, D. (2000). The trade-off between intracell and intercell moves in group technology cell formation. *Journal of Manufacturing Systems*, 19 (5), 305-317.

Aljaber, N., Baek W. & Chen, C.-L. (1997). A tabu search approach to the cell formation problem. *Computers and Industrial Engineering*, 32 (1), 169–185.

Al-Sultan, K.S. & Fedjki, C.A. (1997). A genetic algorithm for the part family formation problem. *Production Planning & Control*, 8 (8), 788-796.

Andres, C. & Lozano, S. (2006). A particle swarm optimization algorithm for part–machine grouping. *Robotics and Computer-Integrated Manufacturing*, 22, 468–474.

Anvari, M., Mehrabad, M.S. & Barzinpour, F. (2010). Machine–part cell formation using a hybrid particle swarm optimization. *International Journal of Advanced Manufacturing Technology*, 47, 745-754.

Arkat, J., Saidi, M. & Abbasi, B. (2007). Applying simulated annealing to cellular manufacturing system design. *International Journal of Advanced Manufacturing Technology*, 32, 531-536.

Ateme-Nguema, B., H. & Dao, T.,-M. (2007). Optimization of cellular manufacturing systems design using the hybrid approach based on the ant colony and tabu search techniques. *Proceedings of the IEEE IEEM*, 668-673.

Ateme-Nguema, B., H. & Dao, T.,-M. (2009). Quantized Hopfield networks and tabu search for manufacturing cell formation problems. *International Journal of Production Economics*, 121, 88-98.

Bajestani, M.A., Rabbani, M., Rahimi-Vahed, A.R. & Khoshkhou, G.B. (2009). A multi-objective scatter search for a dynamic cell formation problem. *Computers & Operations Research*, 36, 777–794.

Baykasoglu, A., Gindy, N.N.Z. & Cobb R. C. (2001). Capability based formulation and solution of multiple objective cell formation problems using simulated annealing. *Integrated Manufacturing Systems*, 12 (4), 258-275.

Bianchi, L., Dorigo, M., Gambardella, L.M. & Gutjahr, W.J. (2009). A survey on meta-heuristics for stochastic combinatorial optimization. *International Journal of Natural Computing*, 8 (2), 239-287.

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

Boulif, M. & Atif. K. (2006). A new branch-&-bound-enhanced genetic algorithm for the manufacturing cell formation problem. *Computers & Operations Research*, 33, 2219–2245.

Boulif, M. & Atif. K. (2008). A new fuzzy genetic algorithm for the dynamic bi-objective cell formation problem considering passive and active strategies. *International Journal of Approximate Reasoning*, 47, 141-165.

Brown, E.C. & Sumichrast, R.T. (2001). CF-GGA: a grouping genetic algorithm for the cell formation problem, *International Journal of Production Research*, 39 (16), 3651-3669.

Burbidge, J.L. (1963). Production flow Analysis. *Production Engineer*, 42 (12), 742-752.

Cao, D. & Chen, M. (2004). Using penalty function and tabu search to solve cell formation problems with fixed cell cost. *Computers & Operations Research*, 31, 21–37.

Cao, D., Defersha, F. M. & Chen, M. (2009). Grouping operations in cellular manufacturing considering alternative routings and the impact of run length on product quality. *International Journal of Production Research*, 47 (4), 989-1013.

Caprihan, R., Slomp, J., Gursaran & Agarwal, K. (2009). A quantum particle swarm optimization approach for the design of virtual manufacturing cells. *Proceedings of the IEEE IEEM*, 125-129.

Car, Z. and Mikac, T. (2006). Evolutionary approach for solving cell-formation problem in cell manufacturing. *Advanced Engineering Informatics*, 20, 227–232.

Caux, C., Bruniaux, R. & Pierreval, H. (2000). Cell formation with alternative process plans and machine capacity constraints: A new combined approach. *International Journal of Production Economics*, 64, 279-284.

Chan, F. T. S., Lau, K. W. & Chan, P. L. Y. (2004). A holistic approach to manufacturing cell formation: incorporation of machine flexibility and machine aggregation. *Journal of Engineering Manufacture*, 218 (B), 1279-1296.

Chan, F.T.S., Lau, K.W., Chan, P.L.Y. & Choy, K.L. (2006). Two-stage approach for machine-part grouping and cell layout problems. *Robotics and Computer-Integrated Manufacturing*, 22, 217–238.

Chan, F.T.S., Lau, K.W., Chan, L.Y. & Lo, V.H.Y. (2008). Cell formation problem with consideration of both intracellular and intercellular movements, *International Journal of Production Research*, 46 (10), 2589–2620.

Chen, W.-H. & Srivastava, B. (1994). Simulated annealing procedures for forming machine cells in group technology. *European Journal of Operational Research*, 75 (1), 100-111.

Chi, S.-C. & Lin, I. (2002). Cellular formation based on evolutionary optimization of granules. *Proceedings of IIE Annual Conference*, Norcross, 1-6.

Chi, S.-C. & Yan, M.-C. (2004). A fuzzy genetic algorithm for high-tech cellular manufacturing system design. *IEEE Annual Meeting of the Fuzzy Information*, 2, 907-912.

Chu, C.-H. & Chang-Chun-Tsai. (2001). A Heuristic Genetic Algorithm for Grouping Manufacturing Cells, *IEEE Proceedings of the 2001 congress on Evolutionary Computation*, 1, 310-317.

Das, K., Lashkari, R. S. & Sengupta, S. (2006). Reliability considerations in the design of cellular manufacturing systems a simulated annealing-based approach. *International Journal of Quality & Reliability Management*, 23 (7), 880-904.

Darwin, C. (1929). *The Origin of Species by Means of Natural Selection or the Preservation of Favored Races in the Struggle for Life*. The Book League of America, (originally published in 1859).

Defersha, F. M. & Chen, M. (2006). Machine cell formation using a mathematical model and a genetic-algorithm-based heuristic. *International Journal of Production Research*, 44 (12), 2421–2444.

Defersha, F. M. & Chen, M. (2008a). A linear programming embedded genetic algorithm for an integrated cell formation and lot sizing considering product quality. *European Journal of Operational Research* 187, 46–69.

Defersha, F. M. & Chen, M. (2008b). A parallel genetic algorithm for dynamic cell formation in cellular manufacturing systems. *International Journal of Production Research*, 46 (22), 6389–6413.

Defersha, F. M. & Chen, M. (2008c). A parallel multiple Markov chain simulated annealing for multi-period manufacturing cell formation problems. *International Journal of Advanced Manufacturing Technology*, 37, 140-156.

Deljoo, V., Al-e-hashem, S.M.J.M., Deljoo, F. & Aryanezhad, M.B. (2010). Using genetic algorithm to solve dynamic cell formation problem. *Applied Mathematical Modelling*, 34, 1078–1092.

Dimopoulos, C. (2006). Multi-objective optimization of manufacturing cell design. *International Journal of Production Research*, 44 (22), 4855-4875.

Dimopoulos, C. & Mort, N. (2001). A hierarchical clustering methodology based on genetic programming for the solution of simple cell-formation problems. *International Journal of Production Research*, 39 (1), 1-19.

Dorigo, M. & Stutzle, T. (2004). *Ant Colony Optimization*. MIT Press, Cambridge, MA, USA.

Durán, O., Rodriguez, N. & Consalter, L.A. (2010). Collaborative particle swarm optimization with a data mining technique for manufacturing cell design. *Expert Systems with Applications*, 37, 1563–1567.

Fan, J., Cao, M. & Feng, D. (2010). Multi-objective dual resource-constrained model for cell formation problem. *Proceedings of the IEEE ICMIT*, 1031-1036.

Filho, E. V. G. & Tiberti, A. J. (2006). A group genetic algorithm for the machine cell formation problem. *International Journal of Production Economics*, 102, 1–21.

Fisher, R. A. (1930). *The genetical theory of natural selection*. Oxford: Clarendon Press.

Foulds, L. R., French, A. P. & Wilson, J. M., (2006). The sustainable cell formation problem: manufacturing cell creation with machine modification costs. *Computers & Operations Research*, 33, 1010–1032.

Giri, R., Srinivas, J. & Mouli, K. V. V. C. (2007). An optimal design approach for a cellular manufacturing system. *Journal of Engineering Manufacture*, 22 (B), 1101-1106.

Glover, F. & Laguna, M. (1997). *Tabu Search*. Kluwer Academic Publishers, Norwell, MA, USA.

Goldberg, D. E. (1989). *Genetic Algorithms in Search Optimization and Machine Learning*. Addison Wesley.

Goncalves, J.F. & Resende, M.G.C. (2004). An evolutionary algorithm for manufacturing cell formation. *Computers & Industrial Engineering*, 47, 247–273.

Gravel, M., Nsakanda, A. L. & Price, W. (1998). Efficient solutions to the cell-formation problem with multiple routings via a double-loop genetic algorithm. *European Journal of Operational Research*, 109, 286-298.

Gupta, Y., Gupta, M., Kumar, A. & Sundaram, C. (1996). A genetic algorithm-based approach to cell composition and layout design problems. *International Journal of Production Research*, 34 (2), 447-482.

Haleh, H., Iranmanesh, H. & Kor, H. (2009). A new hybrid evolutionary algorithm for solving multi objective cell formation problem. *IEEE International Conference on Computers & Industrial Engineering*, 612-616.

Hsu, C.-M. & Su. C.-T. (1998). Multi-objective machine-component grouping in cellular manufacturing: a genetic algorithm. *Production Planning & Control*, 9 (2), 155-166.

Hu, L. & Yasuda, K. (2005). Minimising material handing cost in cell formation with alternative processing routes by grouping genetic algorithm. *International Journal of Production Research*, 44 (11), 2133-2167.

Hwang, H. & Sun, J.-U. (1996). A genetic algorithm-based heuristic for the GT cell formation problem. *Computers & Industrial Engineering*, 30 (4), 941-955.

Islier, A. A. (2005). Group technology by an ant system algorithm. *International Journal of Production Research*, 43 (5), 913-932.

James, T. L., Brown, E. C. & Keeling, K. B. (2007). A hybrid grouping genetic algorithm for the cell formation problem. *Computers & Operations Research*, 34, 2059–2079.

Jayaswal, S. & Adil, G. K. (2004). Efficient algorithm for cell formation with sequence data, machine replications and alternative process routings. *International Journal of Production Research*, 42 (12), 2419-2433.

Joines, J.A., Culbreth, C.T. & King, R.E. (1996). Manufacturing cell design: an integer programming model employing genetic algorithms. *IIE Transactions*, 28 (1), 69–85.

Kao, Y. & Li, Y.L. (2008). Ant colony recognition systems for part clustering problems. *International Journal of Production Research*, 46 (15), 4237-4258.

Kao, Y., Lin, J.-C., & Wu, J-.K., (2008). A Differential Evolution Approach for Machine Cell Formation. *Proceedings of the IEEE IEEM*, 772-775.

Kazerooni, M. L., Luong, H. S. & Abhary, K. (1997). A genetic algorithm based cell design considering alternative routing. *International Journal of Computer Integrated Manufacturing Systems*, 10 (2), 93–107.

Kennedy, J. & Eberhart, R. C. (1995). Particle swarm optimization. *In: Proceedings of IEEE international conference on neural networks*, 1942–1948.

Kor, H., Iranmanesh, H., Haleh, H. & Hatefi, S.M. (2009). A multi-objective genetic algorithm for optimization of cellular manufacturing system. *International Conference on Computer Engineering and Technology*, 1, 252-256.

Lee-Post, A. (2000). Part family identification using a simple genetic algorithm. *International Journal of Production Research*, 38 (4), 793-810.

Lei, D. & Wu, Z. (2006). Tabu search for multiple-criteria manufacturing cell design. *International Journal of Advanced Manufacturing Technology*, 28, 950-956.

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

Liu, C.-M. & Wu, J.K. (1993). Machine cell formation: using the simulated annealing algorithm. *International Journal of Computer Integrated Manufacturing*, 6 (6), 335-349.

Logendran, R. & Karim, Y. (2003). Design of manufacturing cells in the presence of alternative cell locations and material transporters. *Journal of the Operational Research Society*, 54, 1059–1075.

Logendran, R., Ramakrishna, P. & Sriskandarajah, C. (1994). Tabu search-based heuristics for cellular manufacturing systems in the presence of alternative process plan. *International Journal of Production Research,* 32 (2), 273-297.

Lozano, S., Adenso-Diaz, B., Eguia, I. & Onieva, L. (1999). A one-step tabu search algorithm for manufacturing cell design. *Journal of the Operational Research Society*, 50, 509–516.

Mahapatra, S.S. & Sudhakara Pandian, R. (2008). Genetic cell formation using ratio level data in cellular manufacturing systems. *International Journal of Advanced Manufacturing Technology*, 38, 630-640.

Mahdavi, I., Paydar, M. M., Solimanpur, M. & Heidarzade, A. (2009). Genetic algorithm approach for solving a cell formation problem in cellular manufacturing. *Expert Systems with Applications*, 36, 6598–6604.

Mahesh, O. & Srinivasan, G. (2006). Multi-objectives for incremental cell formation problem. *Annals of Operation Research*, 143, 157-170.

Mak, K. L., Peng, P., Wang, X. X. & Lau, T.L. (2007). An ant colony optimization algorithm for scheduling virtual cellular manufacturing systems. *International Journal of Computer Integrated Manufacturing*, 20 (6), 524 – 537.

Mak, K. L. & Wong, Y. S. (2000). Genetic design of cellular manufacturing systems. *Human Factors and Ergonomics in Manufacturing*, 10 (2), 177–192.

Mak, K. L., Wong, Y. S. & Wang, X. X. (2000). An Adaptive Genetic Algorithm for Manufacturing Cell Formation. *The International Journal of Advanced Manufacturing Technology*, 16 (7), 491-497.

Mansouri, S. A., Moattar-Husseini, S. M. & Zegordi, S. H. (2003). A genetic algorithm for multiple objective dealing with exceptional elements in cellular manufacturing. *Production Planning & Control*, 14 (5), 437-446.

Megala, N., Rajendran, C. & Gopalan, R. (2008). An ant colony algorithm for cell-formation in cellular manufacturing systems. *European Journal of Industrial Engineering*, 2 (3), 298-335.

Mehdizadeh, E. & Tavakkoli-Moghaddam, R. (2009). A fuzzy particle swarm optimization algorithm for a cell formation problem. *Proceedings of IFSA-EUSFLAT*, 1768-1772.

Ming, L.C. & Ponnambalam, S.G. (2008). A hybrid GA/PSO for the concurrent design of cellular manufacturing system. *IEEE International Conference on Systems, Man and Cybernetics*, 1855-1860.

Moon, C. & Gen, M. (1999). A genetic algorithm-based approach for design of independent manufacturing cells. *International Journal of Production Economics*, 60–61, 421–426.

Morad, N. & Zalzala, A. M. S. (1996). Formulations for cellular manufacturing and batch scheduling using genetic algorithms. *UKACC International Conference on CONTROL*, 1(427), 473-478.

Murthy, C.V.R. & Srinivasan, G. (1995). Fractional cell formation in group technology. *International Journal of Production Research*, 33 (5), 1323-1337.

Muruganandam, A., Prabhaharan, G., Asokan, P. & Baskaran, V. (2005). A memetic algorithm approach to the cell formation problem. International *Journal of Advanced Manufacturing Technology*, 25, 988–997.

Nair G.J. & Narendran, T. T. (1999). ACCORD: a bicriterion algorithm for cell formation using ordinal and ratio-level data. *International Journal of Production Research*, 37 (3), 539-556.

Neto, A. R. P. & Filho, E. V. G. (2010). A simulation-based evolutionary multiobjective approach to manufacturing cell formation. *Computers & Industrial Engineering*, 59, 64–74.

Noktehdan, A., Karimi, B. & Kashan, A. H. (2010). A differential evolution algorithm for the manufacturing cell formation problem using group based operators. *Expert Systems with Applications*, 37 (7), 4822-4829.

Nsakanda, A. L., Diaby, M. & Price, W. L. (2006). Hybrid genetic approach for solving large-scale capacitated cell formation problems with multiple routings. *European Journal of Operational Research*, 171, 1051–1070.

Onwubolu, G. C. Mutingi, M. (2001). A genetic algorithm approach to cellular manufacturing systems. *Computers & Industrial Engineering*, 39, 125-144.

Onwubolu, G. C. & Songore, V. (2000). A tabu search approach to cellular manufacturing systems. *Production Planning & Control*, 11 (2), 153-164.

Pai, P.F., Chang, P.-T & Lee, S.-H. (2005). Part-machine family formation using genetic algorithms in a fuzzy environment. *International Journal Advanced Manufacturing Technology*, 25, 1175–1179.

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

Papaioannou, G. & Wilson, J. M. (2009). Fuzzy extensions to integer programming models of cell-formation problems in machine scheduling. *Annals of Operations Research*, 166 (1), 163-181.

Papaioannou, G. & Wilson, J. M. (2010). The evolution of cell formation problem methodologies based on recent studies (1997–2008): Review and directions for future research. *European Journal of Operational Research*, 206 (3), 509-521.

Pham, D. T., Afify, A. & Koc, E. (2007). Manufacturing cell formation using the bees algorithm. *IPROMS 2007 Innovative Production Machines and Systems Virtual Conference*, Cardiff, UK.

Pham, D. T., Ghanbarzadeh, A., Koç, E., Otri, S., Rahim, S. & Zaidi, M. (2006). The bees algorithm – A novel tool for complex optimization problems, *Proceedings of IPROMS Conference*, 454–461.

Pierreval, H. & Plaquin, M.-F. (1998). An evolutionary approach of multi criteria manufacturing cell formation. *International Transactions in Operational Research*, 5 (1), 13-25.

Pillai, V. M. & Subbarao, K. (2007). A robust cellular manufacturing system design for dynamic part population using a genetic algorithm. *International Journal of Production Research*, 46 (18), 5191-5210.

Plaquin, M.-F. & Pierreval, H. (2000). Cell formation using evolutionary algorithms with certain constraints. *International Journal of Production Economics*, 64, 267-278.

Ponnambalam, S. G., SudhakaraPandian, R., Mohapatra, S.S. & Saravanasankar, S. (2007). Cell formation with workload data in cellular manufacturing system using genetic algorithm. *Proceedings of the IEEE IEEM*, 674-678.

Prabhaharan, G., Muruganandam, A., Asokan, P. & Girish, B. S. (2005). Machine cell formation for cellular manufacturing systems using an ant colony system approach. *International Journal of Advanced Manufacturing Technology*, 25, 1013–1019.

Rajagopalan, R. & Fonseca, D. J. (2005). Volume sensitivity analysis for manufacturing cells: A genetic algorithm approach. *Journal of Advanced Manufacturing Systems*, 4 (2), 167–183.

Rajagopalan, R. & Fonseca, D. J. (2006). A genetic algorithm approach for machine cell formation. *Journal of Advanced Manufacturing Systems*, 5 (1), 27-44.

Rodrigues, L. C. A. & Weller, T. R. (2008). Cell Formation with Alternative Routings and Capacity Considerations: A Hybrid Tabu Search Approach. *Lecture Notes in Computer Science, MICAI 2008*: 5317, 482-491.

Rogers, D. F. & Kulkarni, S. S. (2005). Optimal bivariate clustering and a genetic algorithm with an application in cellular manufacturing. *European Journal of Operational Research*, 160, 423–444.

Safaei, N., Saidi-Mehrabad, M. & Jabal-Ameli, M.S. (2008). A hybrid simulated annealing for solving an extended model of dynamic cellular manufacturing system, *European Journal of Operational Research*, 185, 563–592.

Schaller, J. (2005). Tabu search procedures for the cell formation problem with intra-cell transfer costs as a function of cell size. *Computers & Industrial Engineering*, 49, 449–462.

Selim, M. S., Askin, R. G. & Vakharia, A. J. (1998). Cell formation in group technology: review evaluation and directions for future research. *Computers & Industrial Engineering*, 34 (1), 3-20.

Sofianopoulou, S. (1999). Manufacturing cells design with alternative process plans and/or replicate machines. *International Journal of Production Research*, 37 (3), 707-720.

Solimanpur, M., Saeedi, S. & Mahdavi, I. (2010). Solving cell formation problem in cellular manufacturing using ant-colony-based optimization. *International Journal of Advanced Manufacturing Technology*, DOI 10.1007/s00170-010-2587-5.

Solimanpur, M., Vrat, P. & Shankar, R. (2004). A multi-objective genetic algorithm approach to the design of cellular manufacturing systems. *International Journal of Production Research*, 42 (7), 1419-1441.

Souilah, A. (1995). Simulated annealing for manufacturing systems layout design. *European Journal of Operational Research*, 82, 592-614.

Spiliopoulos, K. & Sofianopoulou, S. (2003). Designing manufacturing cells: a staged approach and a tabu search algorithm. *International Journal of Production Research*, 41 (11), 2531–2546.

Spiliopoulos, K. & Sofianopoulou, S. (2008). An efficient ant colony optimization system for the manufacturing cells formation problem. *International Journal of Advanced Manufacturing Technology*, 36, 589–597.

Su, C.-T. & Hsu, C.-M. (1998). Multi-objective machine-part cell formation through parallel simulated annealing. *International Journal of Production Research*, 36 (8), 2185-2207.

Sudhakarapandian R. (2007). Application of soft computing techniques for cell formation considering operational time and sequence. PhD thesis, submitted at NIT Rourkela, India.

Suer, G. A. (1997). Evolutionary Programming for Designing Manufacturing Cells. *IEEE International Conference on Evolutionary Computation*, 379-384.

Sun, D., Lin, L. & Batta, R. (1995). Cell formation using tabu search. *Computers & Industrial Engineering*, 28 (3), 485-494.

Tariq, A., Hussain, I. & Ghafoor, A. (2009). A hybrid genetic algorithm for machine-part grouping. *Computers & Industrial Engineering*, 56, 347–356.

Tavakkoli-Moghaddam, R., Aryanezhad, M. B., Safaei, N. & Azaron, A. (2005). Solving a dynamic cell formation problem using meta-heuristics. *Applied Mathematics and Computation*, 170, 761–780.

Tavakkoli-Moghaddam, R., Aryanezhad, M.B., Safaei, N., Vasei, M. & Azaron, A. (2007). A new approach for the cellular manufacturing problem in fuzzy dynamic conditions by a genetic algorithm. *Journal of Intelligent & Fuzzy Systems*, 18, 363–376.

Tavakkoli-Moghaddam, R., Rahimi-Vahed, A.R., Ghodratnama, A. & A. Siadat, A. (2009). A simulated annealing method for solving a new mathematical model of a multi-criteria cell formation problem with capital constraints, *Advances in Engineering Software*, 40, 268–273.

Tavakkoli-Moghaddam, R., Safaei, N. & Sassani, F. (2008). A new solution for a dynamic cell formation problem with alternative routing and machine costs using simulated annealing. *Journal of the Operational Research Society*, 23, 916–924.

Tunnukij, T. & Hicks, C. (2009). An enhanced grouping genetic algorithm for solving the cell formation problem. *International Journal of Production Research*, 47 (7), 1989-2007.

Vakharia, A. J. & Chang, Y.-L. (1997). Cell formation in group technology: a combinatorial search approach. *International Journal of Production Research*, 35 (7), 2025-2043.

Venugopal, V. & Narendran, T. T. (1992). Cell formation in manufacturing systems through simulated annealing: An experimental evaluation. *European Journal of Operational Research*, 63 (3), 409-422.

Venugopal, V. & Narendran, T.T. (1992). A genetic algorithm approach to the machine-component grouping problem with multiple objectives. *Computers & Industrial Engineering*, 22 (4), 469-480.

Vin, E., Lit, P.D. & Delchambre, A. (2005). A multi-objective grouping genetic algorithm for the cell formation problem with alternative routings. *Journal of Intelligent Manufacturing*, 16, 189–205.

Wu, T.-H., Chang, C.-C. & Yeh, J.-Y. (2007). A hybrid heuristic algorithm adopting both Boltzmann function and mutation operator for manufacturing cell formation problems. *International Journal of Production Economics*, 120, 669–688.

Wu, T.-H., Chung, S.-H. & Chang, C.-C. (2008). A simulated annealing algorithm for manufacturing cell formation problems. *Expert Systems with Applications*, 34, 1609–1617.

Wu, T.-H., Chung, S.-H. & Chang, C.-C. (2009). Hybrid simulated annealing algorithm with mutation operator to the cell formation problem with alternative process routings. *Expert Systems with Applications*, 36, 3652–3661.

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

Wu, T.-H., Low, C. & Wu, W.-T. (2004). A tabu search approach to the cell formation problem. *International Journal of Advanced Manufacturing Technology*, 23, 916-924.

Wu, T.-H., Yeh, J.-Y. & Chang, C.-C. (2007). A hybrid simulated annealing algorithm to the cell formation problem with alternative process plans, *International Conference on Convergence Information Technology*, 199-203.

Wu, T.-H., Yeh, J.-Y. & Chang, C.-C. (2009). A hybrid tabu search algorithm to cell formation problem and its variants. World Academy of Science, *Engineering and Technology*, 53, 1090-1094.

Wu, X., Chu, C.-H. & Wang, Yan, W. (2002). A genetic algorithm for integrated cell formation and layout decisions. *IEEE Proceedings of the 2002 congress on Evolutionary Computation*, 2, 1866-1871.

Wu, X., Chu, C.-H. &Wang, Yan, W. (2006). Concurrent design of cellular manufacturing systems: a genetic algorithm approach. *International Journal of Production Research*, 44 (6), 1217-1241.

Wu, X., Chu, C.-H., Wang & Yan, W. (2007). A genetic algorithm for cellular manufacturing design and layout. *European Journal of Operational Research*, 181, 156–167.

Xambre, A.R. & Vilarinho, P.M. (2003). A simulated annealing approach for manufacturing cell formation with multiple identical machines. *European Journal of Operational Research*, 151, 434–446.

Xing, B., Gao, W.-J., Nelwamondo, F.V., Battle, K. & Marwala, T. (2010). Part-Machine Clustering: The Comparison between Adaptive Resonance Theory Neural Network and Ant Colony System. *Lecture Notes in Electrical Engineering, Advances in Neural Network Research and Applications*, 67 (8), 747-755.

Yang, F.-C. & Wang, Y.-P. (2007). Water flow-like algorithm for object grouping problems. *Journal of the Chinese Institute of Industrial Engineers*, 24 (6), 475–488.

Yasuda, K., Hu, L. & Yin, Y. (2005). A grouping genetic algorithm for the multi-objective cell formation problem. *International Journal of Production Research*, 43 (4), 829-853.

Zhao, C. & Wu, Z. (2000). A genetic algorithm for manufacturing cell formation with multiple routes and multiple objectives. *International Journal of Production Research*, 38 (2), 385-395.

Zhao, L., Tsujimura, Y. & Gen, M. (1996). Genetic algorithm for fuzzy clustering. *Proceedings of IEEE International Conference on Evolutionary Computation*, 716-719.

Zhou, M. & Askin, R.G. (1998). Formation of general GT cells: an operation based approach. *Computers & Industrial Engineering*, 34 (1), 147-157.

Zolfaghari, S. & Liang, M. (1998). Machine cell/part family formation considering processing times and machine capacities: a simulated annealing approach. *Computers & Industrial Engineering*, 34 (4), 813-823.

Zolfaghari, S. & Liang, M. (2003). A new genetic algorithm for the machine/part grouping problem involving processing times and lot sizes. *Computers & Industrial Engineering*, 45, 713–731.

Zolfaghari, S. & Liang, M. (2004). Comprehensive machine cell/part family formation using genetic algorithms. *Journal of Manufacturing Technology Management*, 15 (6), 433-444.