Comparison between simultaneous and iterative resolutions for generalized cell formation problems with a Genetic Algorithm

Emmanuelle Vin* and Alain Delchambre*

**BEAMS Department, Université libre de Bruxelles (ULB), Brussels, B-1050, Belgium.*
(Tel.: +(32) 2650-2814; e-mail: Emmanuelle.vin@ulb.ac.be)

Abstract: Actually, lean manufacturing is a must to improve productivity and reduce cost. One of the tools of lean is cellular manufacturing. This technique subdivides the factory to several small entities, which are easier to manage. The algorithm proposed in this paper is based on the simultaneous resolution of two interdependent problems. These two problems emerge when the notion of flexibility is introduced in the production process. This paper demonstrates the efficiency of the simultaneous resolution through a unique adapted genetic algorithm compared to a sequential resolution with iterations.

Keywords: Cellular Manufacturing, Genetic Algorithm, Flow Analysis.

1. INTRODUCTION

For each manager in industry, lean manufacturing is an essential method to improve productivity and reduce cost. Several tools, such as just-in-time, kanban, or SMED, are known to achieve lean objectives,. A less well-known tool, cellular manufacturing, implies a new organization as well as a new mindset.

Cellular production systems are an important application of group technology, which consists of decomposing systems into sub-systems, as well as grouping components together. Cellular manufacturing systems are based on the creation and management of several production cells. Complementary machines placed as close as possible to each other, and dedicated to a specific product family, compose these cells. One of the steps towards the implementation of these cells is precisely to define them, i.e. to resolve the **cell formation problem**.

Initially, the cell formation problem, and the methods used to create product families were relatively simple. Over time, the problem has evolved and become more difficult with data complexity. Progressively, new production parameters have been introduced, such as sequence operation, cost, alternative process plans, part volumes, machine capacity, labour-related factors, and flexibility. Vin (2010) proposes a classification and a complete review based on these parameters used for a more complex cell formation problem.

During the last several years, the cell formation problem has been addressed in numerous works. Several methods have been presented and can be classified in different ways. Joines et al. (1996) presents a complete review of production oriented manufacturing cell formation techniques.

This paper is organized as follow: in section 2, we introduce the concept of flexibility in the cell formation problem and we present a summary table of attributes addressed by authors when flexibility is used. Section 3 describes the problem with all used parameters and constraints, followed by the mathematic formulation in section 4. The proposed method, based on the grouping genetic algorithm (GGA), and the different characteristics of the implementation are explained in section 5. A comparison of the methods is presented in section 6, followed by our conclusion.

2. USE OF FLEXIBILITY

In the cell formation problem many authors address flexibility, using different definitions and assigning different levels of significance to this term. Kim (1993) considers three types of flexibility in process planning:

• Operation or routing flexibility relates to the possibility of performing an operation on alternative machines, with the possibility of distinct processing times and costs.

• Sequencing flexibility corresponds to the possibility of interchanging sequences in which manufacturing operations are performed.

• Processing flexibility is determined by the possibility of producing the same manufacturing feature with alternative operations or sequences of operations.

The first and the second type, respectively, involve alternative machines and alternative sequences, but leave unchanged the operations to be performed. Allowing these flexibilities can provide better performance in mean time flow, throughput, and machine use. The operation flexibility corresponds to defining a type of machine, containing a set of machines able to perform the operation, i.e. the alternative process routes. The two other ones require the use of alternative process plans.

When these flexibilities are introduced, we refer to the Generalized Cell Formation Problem. The resolution of this problem with alternative process routes and plans is equivalent to a resolution of two interdependent problems. The first problem addresses the allocation of a machine to each specific operation, provided that the machine must be able to complete the operation (RP, Resource Planning

problem). The second grouping problem tries to create independent cells by assigning a cell to each machine (CF, Cell Formation problem).

A combination of Artificial Intelligence methods with another heuristic often leads to a resolution of this generalized problem. The resolution can be sequential, semi-simultaneous or completely simultaneous. The sequential resolution provides a solution for the second problem based on the result of the first problem or conversely. The semi-simultaneous resolution is based on several iterations of the sequential resolution. The simultaneous resolution enables us to optimize both problems simultaneously.

In her thesis, Vin (2010) presents a complete review of the authors solving a Generalized Cell Formation Problem. The distinction is made according to the production data used (production volume PV, operating time OT, operation sequences OS and machine capacities MC), the type of the flexibility (Routing, Process and Sequencing Flexibility) and the type of resolution (Hierarchical/sequential, Iterative, Simultaneous). Two more columns are used to notify if the methodology includes several criteria for the evaluation (MO) and its applicability for large-scale problems (LS). Table 1 presents this classification.

Table. 1. Attributes used in the present study and a sample of published papers.

	PV	OS	OT	MC	RF	PF	SF	R	MO	LS
Adenso-Díaz et al., 2001	-	I	-	-	I	-	-	S	-	I
Baykasoglu et al., 2001	I	I	I	I	I	-	-	S	I	I
Caux, 2000	-	-	-	I	I	-	-	S		I
Chen and Heragu, 1999	-	-	-	-	I	-	-	H	-	I
Choobineh, 1988	I	I	I	I	I	I	-	H		0
Defersha and Chen, 2006	I	I	I	I	I	-	-	S	-	
Gravel et al., 1998	I	-	I	I	I	-	-	I	I	
Heragu and Chen, 1998	I	-	-	I	I	-	-	?	-	0
Hwang and Ree, 1996	-	-	-	-	I	-	-	H	-	-
Jayaswal and Adil, 2004	I	I	I	I	I	-	-	I	-	
Jeon and Leep., 2006	I	-	I	I	I	-	-	H	-	0
Joines et al., 1996	-	I	-	-	I	-	-	I	-	
Kang et Wemmerlov, 1993	-	I	-	I	I	-	-	H	I	
Kazerooni et al., 1997	I	I	I	-	I	-	-	H	-	0
Kusiak, 1987	-	I	-	-	I	-	-	H		0
Kusiak and Wemmerlov, 1992	-	-	-	-	I	-	-	I	I	0
Logendran et al., 1994	I	I	I	I	I	-	-	H		
Lozano et al., 1999	I	I	I	I	I	-	-	H	-	0
Mahdavi et al., 2006	I	I	I	I	I	-	-	I	-	I
Mahesh et al., 2002	I	I	I	I	I	-	-	H	-	
Moon and Gen, 1999	I	-	I	I	I	I	-	S		I
Nagi et al., 1990	I	I	I	I	I	-	-	I		0
Nsakanda at al., 2006	I	-		I	I	I	-	I	-	I
Rajamani et al., 1992	I	-	-	I	I	-	-	S		0
Ramabhatta and Nagi, 1998	I	I	I	I	I	-	-	I		0
Sankaran and Kasilingam, 1990	I	-	-	I	I	-	-	H		
Sofianopoulou, 1999	-	I	-	-	I	I	-	S		0
Solimanpur et al., 2004	I	I	I	I	I	-	-	S	I	I
Uddin and Shanker, 2002	I	-	I	I	I	I	-	I	-	
Vin et al., 2003	I	I	I	I	I	-	-	I	I	I
Won, 2000	-	-	-	-	I	-	-	I	I	0
Wu et al., 2004	-	-	-	-	I	-	-	I	-	
Wu et al., 2009	-	-	-	-	I	-	-	S	-	0
Zhao and Wu, 2000	I	I	I	I	I	-	-	I	I	-
This Study	I	I	I	I	I	I	I	S	I	I

In the table, the symbol "I" indicates that the authors consider the feature, as opposed to the symbol "–" indicating that the feature is not used by the authors. An empty cell means that the authors do not specify this feature. For the last criterion (LS), a "0" signifies that the author(s) enumerate(s) all the routings and/or processes necessary to deal with all alternatives (for instance, in the incidence matrix or a similarity coefficient). The enumeration of all solutions implies that the method tends to be very consuming of computational resources. For this last feature, the hypothesis that the method is not applicable to large scales problems can be reasonably assumed.

In this paper, we solve a cell formation problem with real alternative process plans (i.e. the three flexibilities). We propose an adapted genetic algorithm (SIGGA, Simultaneous resolution by a Grouping Genetic Algorithm) to solve both problems simultaneously:

- the routing selection problem and the allocation of operations to a specific machine, yielding flows between machines (resource planning problem with several constraints and criteria);
- the grouping of machines into independent cells (cell formation problem).

We demonstrate the advantages of choosing the simultaneous resolution by comparing it to the sequential resolution that is based on the same algorithm, with a few modification in parameters.

3. DESCRIPTION OF THE PROBLEM

Notations - Indices

t Machines types index (TM_t=Machine type t). t =1,2,..,n_t
m,n Machines index (M_m=Machine m). m =1,2,..,n_m
i Products index (P_i=Product i). i=1,2,..,n_p
j Process index (Pr_{ij}=Process j of product i). j =1,2,..,npr_i
k Operations index (O_{ijk}=Operation k of process j of product i). k =1,2,..,no_{ij}
c Cells index (C_c=Cell c) c =1,2,.., n_c

Parameters

A_m Availability of machine m.
Q_i Quantity of product i.
T_{ijk} Average operating time of operation O_{ijk}.
T_{ijkm} Operating time if O_{ijk} on machine m.
S_c Maximum number of machines in cell c.

Necessary data and hypotheses are presented hereunder. A machine type has different capabilities in terms of operation types. Each machine m is unique and characterized by an availability parameter A_m, which is equal to its capacity value. Each machine belongs to at least one type and can belong to several types in case of a multi-functional machine.

Each product is defined by a set of processes (Process = a sequence of npr_i operations $\{o_{ij1}, o_{ij2}, \ldots, o_{ij\,npri}\}$). Each operation is defined as being accomplished on one type of machine (lathe, grinding machine, etc.). Therefore each operation can be performed on all machines of a particular type. The duration of each operation can be fixed for the considered machine type (average operating time, T_{ijk}), or particularized to a specific machine (operating time, T_{ijkm}). With these specifications, each product has several potential routings available for a specific process.

4. FORMULATION

4. 1. Decision variables

$x_{ij} = 1$ if process j of product i is used (= 0 otherwise).
$y_{ijkm} = 1$ if operation O_{ijk} is performed on machine m (= 0 otherwise).
$z_{mc} = 1$ if machine m is in cell c (= 0 otherwise).

When the algorithm assigns an operation O_{123} to a specific machine M_5, the variable x_{12} is put to 1 to specify that process *2* (Pr_{12}) of product P_1 is used in the solution. This variable implies that all other variables $x_{1j\neq 2}$ of the same product (P_1) are put to 0. In this case, all operations belonging to $P_{1j\neq 2}$ cannot be used in the grouping solution. To complete this notation, decision variable y_{1235} is also equal to 1. Decision variable z_{mc} is used to compute moves between cells as a function of the assignation of machines in each cell.

4.2 RP Constraints

$$\sum_{j=1}^{npr_i} x_{ij} = 1 \quad \forall i \qquad (1) \qquad \sum_{m=1}^{n_M} y_{ijkm} = 1 \quad \forall i,j,k \qquad (2)$$

$$if\,(x_{ij} = 0) => y_{ijkm} = 0 \quad \forall k,m \qquad (3)$$

$$\sum_{i=1}^{n_p}\sum_{j=1}^{npr_i}\sum_{k=1}^{no_{ij}} Q_i.T_{ijkm}.y_{ijkm} \le A_m \quad \forall m \qquad (4)$$

$$if\,(y_{ijkm} = 1) => Q_i.T_{ijkm}. > 0 \quad \forall i,j,k,m \qquad (5)$$

Constraint (1) represents the process selection. As explained above, only one process can be chosen per product. Constraint (2) describes that each operation can be assigned to only one machine. All operations belonging to a non-used process cannot be allocated to a machine (constraints (2)). The fourth constraint defines the machine's charge. This charge cannot exceed the machine's availability. Constraint (5) imposes that an operation assigned to a specific machine must have a strictly positive operating time. Indeed, if a machine belongs to the machine type linked to a specific operation but cannot perform this operation, the operating time T_{ijkm} of the operation O_{ijk} on the machine M_m will be null.

4.3. CF Constraints

$$\sum_{c=1}^{n_c} z_{mc} = 1 \quad \forall m \qquad (6) \qquad \sum_{m=1}^{n_m} z_{mc} \le S_c \quad \forall c \qquad (7)$$

Constraints (6) and (7) concern the grouping of machines into cells. The first one verifies that all machines used during the process have been grouped. The second one confirms that each cell capacity has not been exceeded. The maximum capacity can differ for each cell.

4.4. Cost Function

The proposed method is a multicriteria method, but this paper focuses on one criterion: the maximization of intracellular moves. Equation (8) represents the move between two machines, m and n. It is computed based on the sum of operating time to perform on the machine n for all products coming from the machine m. This value can be computed when the first part of the chromosome is completed and the first problem is solved.

$$\phi_{mn} = \sum_{i=1}^{n_p}\left(\sum_{j=1}^{npr_i} x_{ij}\cdot\left(\sum_{k=1}^{no_{ij}-1}\left(y_{ijkm}\cdot y_{ij(k+1)n}\right)\cdot\left(Q_i\cdot T_{ij(k+1)n}\right)\right)\right) \qquad (8)$$

$$\Phi_{\text{intracell}} = \sum_{c=1}^{n_c}\left(\sum_{m=1}^{nm_c}\sum_{n=1}^{nm_c}\left(z_{mc}.z_{nc}\right).\phi_{mn}\right) \qquad (9)$$

$$\text{Cost function: } Max\frac{\Phi_{\text{intracell}}}{\Phi_{\text{Total}}} \quad \text{with} \quad \Phi_{\text{Total}} = \sum_{m=1}^{n_M}\sum_{n=1}^{n_M}\phi_{mn} \qquad (10)$$

To compute the equation (9), the second part of the chromosome needs to be completed and a valid solution for the cell assignment problem needs to be found. The intracellular moves into a cell c are the sum of moves between all machines assigned to this cell c. The total intracellular move is the sum of intracellular moves for each cell. The total moves can be different depending on process and routing choices. To compare two solutions and take into account this difference, the criterion to maximize is the relative intracellular movement (10). This criterion is the same as the minimization of the total intercellular movements.

The whole problem is solved with a SIGGA whose flowchart is illustrated in Fig. 1. This algorithm is an adaptation of the Grouping Genetic Algorithm (GGA) explained in the next section.

5. SIGGA

5.1. Origins

The genetic algorithms (GAs) are an optimization technique inspired by the evolution process of living organisms (Holland, 1975). The basic idea is to maintain a population of chromosomes, each chromosome being the encoding (a description or genotype) of a solution (phenotype) of the problem being solved. Chromosome worth is measured by its fitness, which is often simply the objective functional value of the search space point defined by the (decoded) chromosome. Falkenauer (1998) pointed out the weaknesses of standard GAs when applied to grouping problems, and introduced the GGA, which is a GA heavily modified to match the structure of grouping problems. Those are the problems where the aim is to group together members of a set (i.e. find a good partition of the set). The GGA operators (crossover, mutation and inversion) are group-oriented, in order to follow the structure of grouping problems.

Vin et al. (2005) presented a GA in two steps. The used algorithm is based on a semi-simultaneous method. First, a population of RP solution (operation/machine) is initialized in the genetic algorithm. Next, for each solution and at each generation of the GA, a heuristic is applied to complete the chromosome with a valid solution for the second problem

(machine/cell): machine grouping into cells. This method was favourable except in large-scale problems where the heuristic cannot find the best solution. In this method, the operators of the GA were applied only on the RP solution.

5.2. Description of the SIGGA

The SIGGA (Simultaneous resolution by a Grouping Genetic Algorithm) is presented in Fig. 1 (completely described in Vin (2010)). This algorithm is based on a classical GGA. Each chromosome represents a valid solution to both problems: the process selection with the assignment of each operation on a specific machine able to achieve it (**R**esource **P**lanning problem: operation/machine); the grouping of machines into independent cells (**C**ell **F**ormation problem: machine/cell).

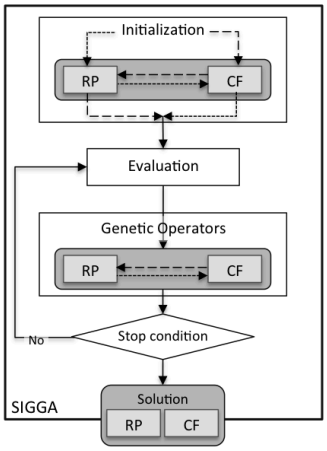

Fig. 1. Adapted SIGGA

Both problems are interdependent because groups (machines) of the RP problem are precisely the objects to group in the CF problem. As shown in Fig. 2, each solution for RP problem is associated to a large search space for the CF problem. Our objective is to find a "good" solution for both problems (RP and CF) in minimizing the intercellular traffic.

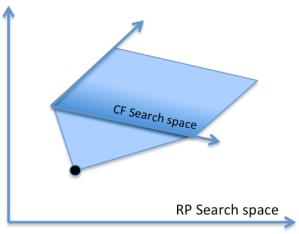

Fig. 2. Embedded search spaces.

A population of chromosomes is initialized. To ensure a population sufficiently diversified; we introduce the initialization rate, ri_1 and ri_2. The parameter ri_1 defines the proportion of the chromosomes that will be initialized following the dotted line where the RP initialization precedes the CF initialization (RP ➔ CF). Conversely, the parameter ri_2 represents the situation in dash line where the RP initialization follows the CF initialization (CF ➔ RP). In our SIGGA, both initialization rates are set to 50%.

A heuristic flow cannot be used to construct the first problem (RP or CF) because there are not flows between the machines yet. A classical First Fit heuristic is used to create this first solution. Once the machines are filled with operations or grouped into cells, a specific heuristic flow is used to complete the second part of the chromosome (CF or RP) by minimizing the intercellular traffic. These heuristics (basic or flow oriented) generate valid solutions respecting all hard constraints defined in section 3.2 and 3.3.

After this initialization stage, the fitness of each chromosome is completed with its evaluation. The best chromosome of the population is saved. In order to evolve to the optimal solution, different genetic operators are applied after a specific selection (tournament strategy). For this step, we introduce the genetic operator rates, ro_1 and ro_2. They express the probability to apply the operators on the first problem (ro_1) or on the second one (ro_2). If $ro_1+ro_2 > 100$, the operators are applied on both parts of the chromosome. In SIGGA, both rates are set to 100%. In this way, the operators are applied to the complete chromosome to make both problems evolve simultaneously. After the application of genetic operators, each chromosome is reconstructed following the same way than the initialisation (RP ➔ CF or CF ➔ RP). The first part is reconstructed with the random heuristic and the second part with a flow oriented heuristic. Then, a new generation is started. The algorithm stops when the maximum number of generation is reached or when the algorithm finds the solution without flow between cells.

6. COMPARISON OF RESOLUTIONS

6.1. Utilization of the SIGGA for the sequential resolutions

The RPGGA is a genetic grouping algorithm addressing the RP problem where all the operations are allocated to a specific machine. To group these machines into cells and take into account the minimization of the intercellular traffic in the evaluation of the RPGGA, the CF-part must be constructed with an integrated module. This integrated module is called to construct the CF-part of the chromosome each time the RP-part is modified, i.e. after the initialization and after all the uses of the genetic operators. During all the treatment by the RPGGA, the genetic operators used to make evolved the population never modify the CF- part.

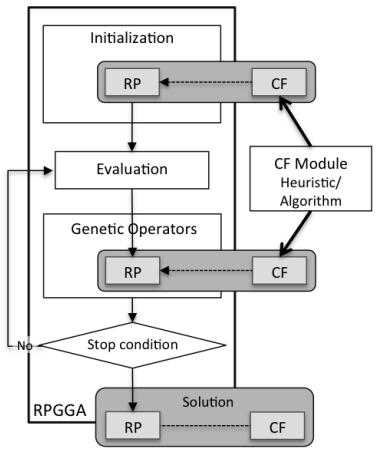

Fig. 3. RPGGA with an integrated CF-module.

In Fig. 3, the flow chart of the RPGGA with an integrated CF-module is shown. This algorithm allows solving both problems (RP and CF) in a sequential way with iterations in the GGA. The module called to construct the second part can be a heuristic or another algorithm/metaheuristic. In this paper, both cases have been tested to construct this second part:

- the RPGGA with an integrated CF heuristic: a good solution for CF-part is found thanks to a specific heuristic flow oriented (*HeurCF* in Fig. 4 and 5);
- the RPGGA with another integrated GGA: to complete each RP-part, an optimal solution for the CF-part is found by a *GGACF* (specific GGA to solve CF problem).

In the latter case, a new GGA is called upon for each individual chromosome in the population and for each modification of the RP-part by the operator. As shown in Fig. 2, the integrated GGA allows exploring the second search space (CF) associated to each solution of the first search space (RP). The integrated heuristic will find a good solution in this search space without exploration.

To use the SIGGA like a sequential resolution (RPGGA), the rates for the first problem (ri_1 and ro_1) are set to 100% and the rates for the second problem (ri_2 and ro_2) are set to 0%. With these parameters, the SIGGA runs in the exactly same way than the RPGGA with the integrated CF-module. Indeed, all the chromosomes will be initialized in the way RP ➔ CF and the operators will be applied only on the RP-parts.

Fig. 3 where each RP is replaced by CF and conversely, represents a CFGGA with an integrated RP-module: heuristic (*HeurRP*) or another GGA (*GGARP*). The explanation is same as previously in reversing all "RP" and "CF".

6.2. Comparison of sequential and simultaneous resolutions

Five resolutions have been compared in terms of number of generations and the time necessary to achieve the best solution. These resolutions have been applied to a large benchmark of case studies. The results presented in this paper are based on an ideal case study whose resolution is possible by creating completely independent cells. In this case, there are 15 types of machines with one to three different machines, 50 products with 2 different processes (operations sequences). The search space for CF problem is evaluated around 10^{15} solutions and 10^{75} for the RP search space.

Different observations can be made about Fig. 4. and Fig. 5.:

- Firstly, there is a significant difference in the evolution of the solutions with the RPGGA (with integrated HeurCF or GGACF) or CFGGA (with integrated HeurRP or GGARP). The explanation lies in the size of the search spaces. The main GGA evolves faster in a small search space (for CF problem).
- Next, in terms of generations, the resolution with an integrated GGA is always faster than the solution with the heuristic. This is due to the fact that the integrated GGA approaches the optimal solution in the second search space while the heuristic sends back only a good solution. However, in terms of time, the use of the GGA greatly delays arrival to the best solution.
- Finally, comparing the resolution with SIGGA and the GGA with another integrated RPGGA, the evolution of the solution is very similar in terms of generation. However, when we look at the time required to achieve this result, there is no possible comparison. Within 20 seconds the SIGGA amounts to 98% of intracellular flow and 100% after only 45 seconds.

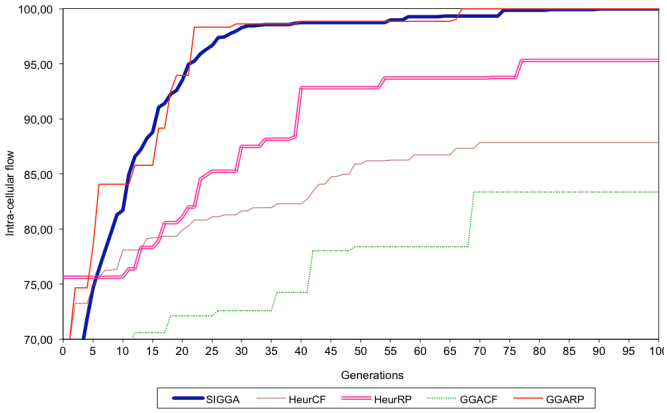

Fig.4. Intracellular flow in function of the generations.

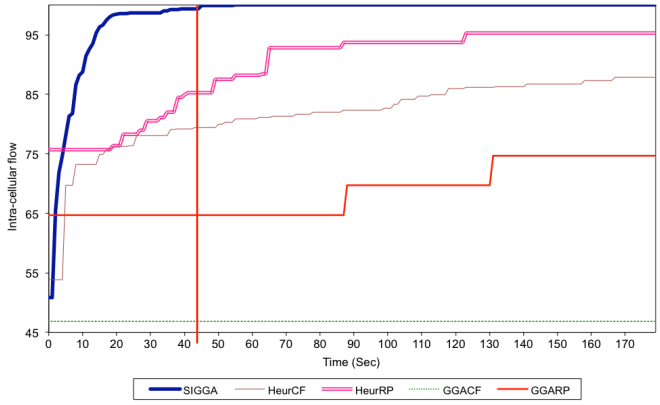

Fig. 5. Evolution of the intracellular flow with the time.

7. CONCLUSIONS

An adapted genetic algorithm is presented to simultaneously solve two interdependent problems in Generalize Cell Formation Problem: the allocation of a specific machine to each operation and the machine grouping into cells. The mathematical model has been presented. This innovative methodology is compared to the iterative resolutions with integrated heuristics or algorithms. These comparisons are based on the same Grouping Genetic Algorithm with identical parameters. Using this method, it is well the resolution (iterative or simultaneous) that is analysed.

REFERENCES

Adenso-Diaz, B., Lozano, S., Racero, J. and Guerrero, F. (2001). Machine cell formation in generalized group technology. *Comput. Ind. Eng.*, 41(2), 227-240.

Baykasoglu, A., Gindy, N.N.Z. and Cobb, R.C. (2001). Capability based formulation and solution of multiple objective cell formation problems using simulated annealing. *Comput. Integr. Manuf. Syst.*, 12, 258–274.

Caux, C. (2000). Cell formation with alternative process plans and machine capacity constraints: A new combined approach. *Int. J. Econ.*, 64, 279–284.

Chen, J.S. and Heragu, S.S. (1999). Stepwise decomposition approaches for large scale cell formation problems. *Eur. J. Op. Res.*, 113, 64–79.

Choobineh, F. (1998). A Framework for the Design of Cellular Manufacturing Systems. *Int. J. Op. Res*, 26(7), 1161-1172.

Defersha, F.M. and Chen, M. (2006). A comprehensive mathematical model for the design of the cellular manufacturing systems. *Int. J. Prod. Econ.*, 103,767–783.

Falkenauer, E. (1998). *Genetic algorithm and grouping problem*. Wiley & Sons.

Gravel, M., Nsakanda, A.L. and Price, W. (1998). Efficient solutions to the cellformation problem with multiple routings via a double-loop genetic algorithm. *Eur. J. Op. Res.*, 109, 286–298.

Heragu, S.S. and Chen, J.S. (1998). Optimal solution of cellular manufacturing system design: Benders' decomposition approach. *Eur. J. Op. Res.*, 107, 175–192.

Holland, J. H. (1975). Adaptation in Natural and Artificial Systems. Ann Arbor: University of Michigan Press.

Hwang, H. and Ree, P. (1996) Routes selection for the cell formation problem with alternative part process plans. *Comp. Ind. Eng.*, 30(3), 423–431.

Jayaswal, S. and Adil, G.K. (2004). An efficient algorithm for cell formation with sequence data, machine replications and alternative process routings. *Int. J. Prod. Res.*, 42(12), 2419–2433.

Jeon, G. and Leep, H.R. (2006). Forming part families by using genetic algorithm and designing machine cells under demand changes. *Comput. Op. Res.*, 33(1), 263–283.

Joines, J., Culbreth, C. and King, R. (1996). A Comprehensive Review of Production Oriented Cell Formation Techniques. *Int. J. Factory Automation and Information Management*, 3(3-4), 225-265.

Kang, S. and Wemmerlov, U. (1993). A work load-oriented heuristic methodology for manufacturing cell formation allowing reallocation of operations. *Eur. J. Op. Res.*, 69(3), 292–311.

Kazerooni, M., Luong, H.S. and Abhary K. (1997). A genetic algorithm based cell design considering alternative routing. *Comput-Integrated Manuf. Syst.*, 10(2), 93–108.

Kim, Y.D. (1993). A study on surrogate objectives for loading a certain type of flexible manufacturing system. *Int. J. Prod. Res.*, 31:381–392.

Kusiak, A. (1987). The Generalised Group Technology Concept. *Int. J. Prod. Res.*, 25(3), 561-569.

Kusiak, A. (1992). *Intelligent Manufacturing Systems*. Prentice Hall, Englewood, NJ.

Logendran, R., Ramakrishna, P. and Srikandarajah, C. (1994). Tabu search-based heuristics for cellular manufaturing systems in the presence of alternative process plans. *Int. J. Prod. Res.*, 32(2), 273-297.

Lozano, S., Guerrero, F., Eguia, I. and Onieva, L. (1999). Cell design and loading in the presence of alternative routing. *Int. J. Op. Res*, 37(14), 3289–3304.

Mahdavi, I., Javadi, B., Fallah-Alipour, K. and Slomp, J. (2007). Designing a new mathematical model for cellular manufacturing system based on cell utilization. *Appl. Math. Comput.*, 190(1), 662-670

Mahesh, O. and Srinivasan, G. (2002). Incremental cell formation considering alternative machines. *Int. J. Op. Res*, 40(14), 3291–3310.

Moon, C. and Gen, M. (1999). A genetic algorithm-based approach for design of independent manufacturing cells. *Int. J. Prod. Econ*, 20(60-1), 421–426.

Nagi, R., Harhalakis, G. and Proth, J.-M. (1990). Multiple routings and capacity considerations in group technology applications. *Int. J. Prod. Res*, 28(12), 2243-2257.

Nsakanda, A., Diaby, M. and Price, W.L. (2006). Hybrid genetic approach for solving large-scale capacitated cell formation problems with multiple routings". *Eur. J. Op. Res.*, 171(3), 1051–1070,

Rajamani, D., Singh, N. and Aneja, Y.P. (1992). Selection of parts and machines for cellularization: a mathematical programming approach. *Eur. J. Op. Res.*, 62(1), 47–54.

Ramabhatta, V. and Nagi, R. (1998). An integrated formulation of manufacturing cell formation with capacity planning and multiple routings. *Ann. Op. Res.*, 77, 79–95.

Sankaran, S. and Kasilingam, G. (1990). An integrated approach to cell formation and part routing in group technology manufacturing systems. *Eng. Optim.*, 16, 235–245.

Sofianopoulou, S. (1999). Manufacturing cell design with alterantive process plans and/or replicate machines. *Int. J. Prod. Res*, 37(3), 707-720.

Solimanpur, M., Vrat, P. and Shankar, R. (2004). A multi-objective genetic algorithm approach to the design of cellular manufacturing systems. *Int. J. Prod. Res.*, 42(7), 1419–1441.

Uddin, M.K. and Shanker, K. (2002). Grouping of parts and machines in presence of alternative process routes by genetic algorithm. *Int. J. Prod. Econ.*, 76(3), 219-228.

Vin, E., DeLit, P.G. and Delchambre, A. (2003). Une approche intégrée pour résoudre le problème de formation des cellules de production avec des routages alternatifs. *In MOSIM03 world symposium*. April 23-25, Toulouse, France.

Vin, E. (2010). *Genetic Algorithm Applied to Generalized Cell Formation Problems*. PhD Thesis, Ecole Polytechnique de Bruxelles, Belgium.

Won, Y. (2000). New p-median approach to cell formation with alternative process plans. *Int. J. Prod. Res*., 38(1), 229–240.

Wu, T.H., Chen, J.F. and Yeh, J.Y. (2004). A decomposition approach to the cell formation problem with alternative process plans. *Int. J. Adv. Manuf. Technol.*, 24(11/12), 834–840.

Wu, T.H., Chung, S.H. and Chang, C.C. (2009). Hybrid simulated annealing algorithm with mutation operator to the cell formation problem with alternative process routing. *Expert Syst. Appl.*, 36, 3652–3661.

Zhao, C. and Wu, Z. (2000). A genetic algorithm for manufacturing cell formation with multiple routes and multiple objectives. *Int. J. Prod. Res.*, 38(2), 385-395.