ISSN 2176-5480

22nd International Congress of Mechanical Engineering (COBEM 2013)
November 3-7, 2013, Ribeirão Preto, SP, Brazil

COMPARATIVE ANALYSIS ON MANUFACTURING CELL FORMATION USING GENETIC ALGORITHMS

Rafael Crespo Izquierdo[1]
Flávio Jose Lorini[2]
Herbert Martins Gomes[3]

Federal University of Rio Grande do Sul (UFRGS) - Departament of Mechanical Engineering (DEMEC), Sarmento Leite St., 425, –Downtown , 90050-170 – Porto Alegre, RS – Brazil.

[1]rcrespo9@hotmail.com ,[2] lorini@ufrgs.br,[3] herbert@mecanica.ufrgs.br

Abstract. *In modern production systems it requires increasing the overall efficiency, which results in the optimization process steps. In this context, the arrangement of an industrial layout is a fundamental concept to the effectiveness of a system that must be considered in its design. An industrial environment can be designed with layouts of several types, according to the diversity of items, adopted production mode and market demand. This paper discusses the design layout of the cellular type, an arrangement used in industrial engineering that allows diversity to meet production compatible with operational flexibility. Among the various techniques and approaches applied to the formation of manufacturing cells, specifically here it is adopted an application of genetic algorithm, implemented using an interface in* Matlab*, where several topics are covered, namely: the generation of the initial population, the chromosome coding, operators of crossover and mutation and reliability of the proposed algorithm. The obtained results show that the Genetic Algorithms are reliable tools for the optimization of manufacturing systems, obtaining responses that meet the basic requirements of a cell design, and presenting convergence values in the generation of machine groups and family of parts.*

Keywords: *Genetic Algorithms, Manufacturing Cells, Industrial Layout*

1. INTRODUCTION

Production systems' increasing productivity demands cause the analysis of the layout to be fundamental, because the physical arrangement of the machines in a manufacturing environment may influence the flow of production and consequently the efficiency and cost of a production process. Nowadays, most production systems´ production is focused on the needs of the customers "just in time" (JIT), because it is a way of avoiding stocks. This leads to the use of cellular layouts, which are more suitable to meet this type of demand. Cellular layouts are widely applied arrangements in engineering industry, for they enable good levels of production with desirable flexibility. In this context various methods can be applied to form cells using specific techniques. This work aims at studying a heuristic method related to the optimization of manufacturing systems through iterative processes, using specific functions and convergence criteria. This paper makes use of information on production flow, with information tabulated in a matrix of incidence part-machine.

This information is treated heuristically, associated with certain restriction conditions, aiming at the diagonalization of incidence part-machine matrices, which consists of the formation of cells by a random combination of machines and parts. The application is intended to define the formation of manufacturing cells, using genetic algorithms. Matlab's toolbox called *gatool* is used for the simulations. In addition, we intend to compare the results for different types of production flow analysis (incidence part-machine matrices), verifying the efficiency of the proposed application with respect to some examples in the literature.

1.1 Designing and Planning of an Industrial Layout

According to Slack *et al.*(1999), the physical arrangement or layout of a production process is related to the physical layout of all the elements needed for the transformation of a specific product.

The importance of planning and decision making in relation to a layout is fundamental to the strategy of a company, because a well prepared project enables a desirable performance with physical arrangements that favor the flexibility of operations and efficiency of flows and use of the resources. Thus, it is necessary to adjust the decision of physical arrangements in accordance with a more competitive strategy of operation. (Corrêa and Corrêa, 2006).

According to Lorenzatto *et al.* (2007), layout optimization eliminates many of the losses related to a determined production process, such as reducing the time of labor involved in transporting materials, increasing productivity and improving quality indices and motivation of employees.

1.2 Types of Layout

According to Slack *et al.*(1999), the basic type of physical arrangement is the general form of productive resources, where there are many ways to arrange these resources. Most of the layouts in manufacturing environments are related to four basic types of physical arrangements, which are the positional layout (fixed position), the line layout (for product), the functional layout (for process) and the cellular layout. The choice of a particular type is often not simple, depending on the diversity of items and volume of production. As illustrated in Figure 1, for a certain range of product variety and volume of production, the characteristics of the physical arrangements overlap, requiring an analysis to determine which type of layout should be chosen, considering the advantages and disadvantages of each type of arrangement. This ratio between range and volume also influences the flow of resources processed, i.e., the greater the variety of products and smaller the production volume, more intermittent becomes the flow. However, the greater the production volumes and the smaller product variety, more continuous becomes the flow.

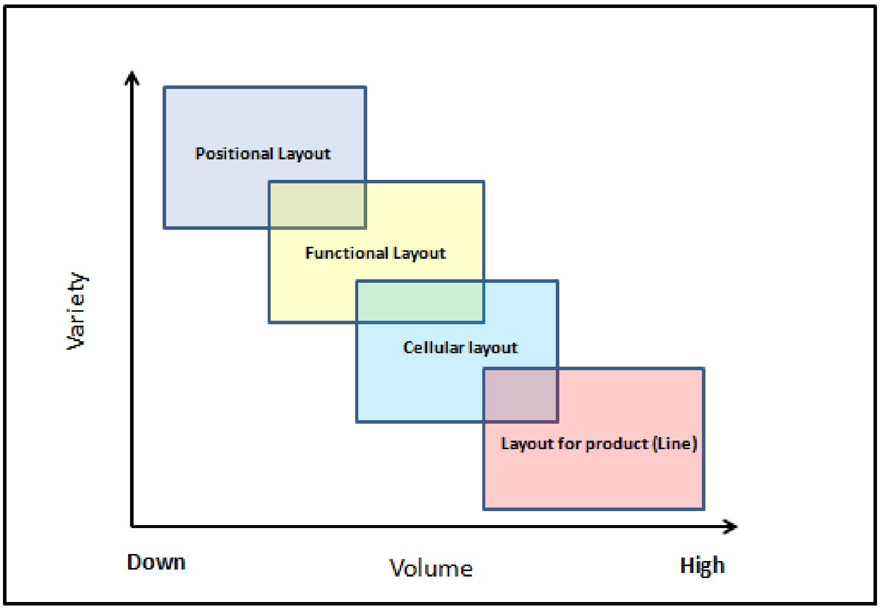

Figure 1. Relationship between variety–volume in the selection of a physic arrangement.

1.3 Cellular Layout

The cellular layout is characterized by the arrangement of machines in groups of various types, not anymore in order to meet the demands of the production of a determined product, but of a family of parts (Lorini, 1993). That is, in this type of physical arrangement the resources processed (when they enter the cycle of operation) are selected according to the constructive characteristics of the product, in order to follow a specific flow within a cell, where the materials, workers and-equipments (resources transformers) needed for the processing of a product are available. (Slack *et al.* al., 1999).

1.4 Techniques of Formation of Cells

Regarding the techniques of formation of cells, the concept of Group Technology (GT) is essential due to its various applications. When related to manufacturing, it consists of philosophy of organization, grouping the parts and resources to the manufacture (Lorini, 1993).

For Groover (2011), there are three general approaches to solve the problem of family of parts grouping. These are methods that require a certain time to implement and involve the analysis of the data by a properly trained staff, namely: (a) Visual inspection, (b) Classification and coding of parts and (c) Production Flow Analysis (PFA).

ISSN 2176-5480

1.4.1 Visual Inspection

The visual inspection method consists of grouping the parts into families, by visual analysis of the similar characteristics of the parts. This is a simple method that does not require large costs for implementation (Groover, 2011). According to Lorini (1993), this method is less efficient, is highly dependent on human experience, and is limited by the number of parts that can be physically tested in an acceptable time. Thefore its application is restricted in the case of small numbers of parts.

1.4.2 Classification and Codification of Parts

A system of classification aims to implement the coding parts, so that process and geometric attributes, or others, are clearly and accurately represented. Thus, once the similarity criteria are established to the parts being encoded based on similarity criteria, the clusters of parts can then be formed. However, this code should not be too complex, otherwise it may end up hindering the formation of families. (Lorini, 1993). According to Groover (2011), the system of classification and coding (SCC) consumes more time than visual inspection and techniques related to production flow analysis (PFA).

1.4.3 Production Flow Analysis (PFA)

The production flow analysis consists of a technique in which the parts are grouped according to the coincidences of process flows set for each component, regardless of size or geometric shape. One way to represent this sequence of operations is through machine-part incidence matrices, establishing the relation part-process in the form of row and columns of a matrix, as shown in Figure 2.

		Parts									
		1	2	3	4	5	6	7	8	9	10
Machines	1	1			1		1			1	
	2		1	1				1			
	3				1	1	1				1
	4		1		1	1				1	1
	5	1		1			1		1		1

Figure 2. Representation of a typical incidence matrix part-machine.

This matrix can be represented in different ways, like symbols (x, *), binary numbers ("0", "1") and integers (0.1, 0.2, ..., n), that characterize the processing time for a specific part and machine.

Algorithms can be applied to the matrix in order to highlight which parts present similarities in their processes, characterizing the formation of a "family", while the group of machines is characterized in the formation of a "cell". There are several types of algorithms that can be applied, e.g.: mathematical programming, hierarchical clustering techniques, techniques based on the arrangement and non-hierarchical clustering techniques (heuristic).

Heuristic techniques, discussed in this paper, consist of non-classificatory procedures that use similarity coefficients, and in most of its applications, they use some kind of criterion of grouping. The heuristic methods are related to a set of rules aimed at solving problems, and these rules depend on the developer's decisions or his/her experience in similar cases. However, a heuristic method does not guarantee the attainment of an optimal solution, but a feasible solution. (Tahara *et al.* 1997).

Among the heuristics and meta-heuristic methods in the cell formation, there can be cited the following ones:

- Singh and Rajamani (1996) comment that the techniques of Artificial Intelligence most known are: NR- Neural Networks, SA- *Simulated Annealing*, TB- *Tabu Search* and GA- *Genetic Algorithm.*

- Arzi *et al.* (2001) propose an application with Genetic Algorithm with an approach for the analysis of the efficiency and capacity requirements in an environment with irregular demand.

- Arkat, *et al.* (2012) propose the study of heuristic models for the study of cell formation, analyzing the algorithms Branch and Bound (B & B) and the Genetic Algorithm.

- Sayadi, *et al.* (2013) propose a model based on the *Firefly* algorithm. This work showed that this method is effective, so that the meta-heuristic should be useful for further studies.

This paper uses the genetic algorithm for the implementation of manufacturing cell formation.

ISSN 2176-5480

2. METHODOLOGY

It will be presented below the structure of the Genetic Algorithm, such as the feature to generate the initial population, the representation of the incidence part-machine matrix, the cost function evaluation, the application of the genetic operators, the stopping criteria, convergence of the algorithm and the interpretation of results.

The computational language Matlab is used for the implemented application and includes specific routines for solving optimization problems by genetic algorithms. In order to deal with the cell formation problem it is necessary to define some specific subroutines. The implementation of this algorithm in the Matlab toolbox is illustrated in the flowchart in the following Figure 3.

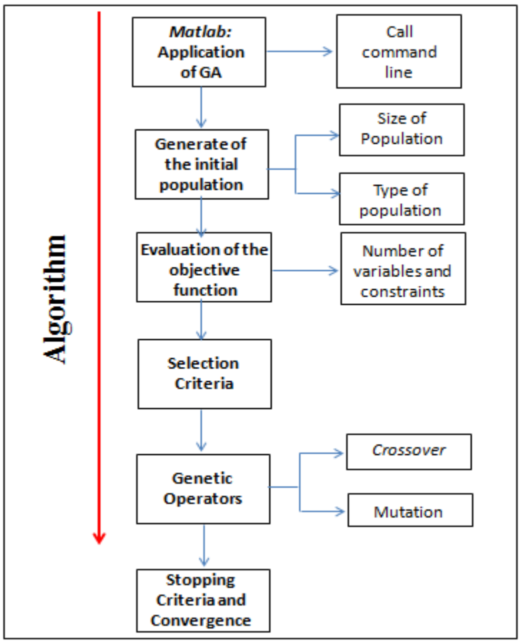

Figure 3. Flowchart of implementation of the algorithm

2.1 Matlab: Application of Genetic Algorithms (GA)

Basically, Matlab allows the use of GA in two ways. The more conventional forms of applications use specific command lines, as indicated in Table 1, or use graphical user interface, which consists of the main interface of GA in Matlab. In the present work, the decision was to use command lines, since this work aims at generating a graphical interface for interaction with the user (the use of command lines facilitates the use of GA).

Table 1. Representation of GA from the command line in Matlab language.

Setting the heuristic parameters of genetic algorithms	
Command lines	**Description**
Population Type: “bitstring”	String used in bits
Population Size	Population composed for 400 individuals
Elite Count	5% of the population is forwarded to the new generation
Crossover Fraction	Crossover rate
Generations	Maximum number of generations
Fitness Scaling “fcn”	Fitness scaling is proportional
Selection “fcn”	The method used is Roulette
Crossover “fcn”	Single Point
Mutation “fcn”	Uniform

ISSN 2176-5480

2.2 Generation of the Initial Population in Cell Formation

Initially the algorithm generates a random population, where the population size is set beforehand and may vary according to the variety of individuals required in the problem to be analyzed. This work is coded according to the chromosome representation (Gonçalves Filho *et al.* 2004), so that each machine has a position in the chromosome, while each character represents a cell. Figure 5 illustrates an example of the GA encoding used in this work.

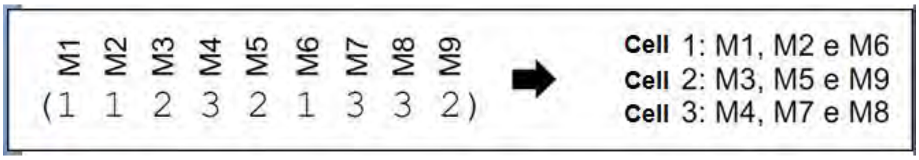

Figure 5. Representation of a chromosome for cell formation.

Considering that the toolbox of Matlab does not work with chromosomes represented by integers, it was created an algorithm that is implemented as subroutine in the toolbox, aiming at performing the conversion of the value entire of the chromosome for binary values.

The population size is related to the number of individuals (chromosomes) which interact during generations. Figure 6 illustrates a population composed of N individuals.

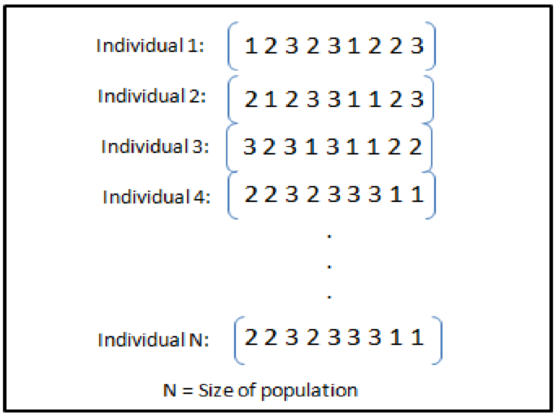

Figure 6. Representation of a population consisting of N individuals.

2.3 Objective Function

Among the existing optimization studies for the formation of the cells, it is observed that the objective function is one of the most discussed items. The idea is to maximize or minimize a problem related to a specific goal.

According to Vernugopal and Naredran (1992), there are several goals that are important and are associated with the problem of machine grouping, but in the case of cell formation, it is important to work with a manageable number of goals that are distinct and measurable. This paper addresses the following objectives:

F1 - minimizing the amount of movement between cells: this reduces material handling, and significantly simplify the control of the factory floor reducing the flow time average and WIP (*Work in process*).

F2 - minimizing the variation of the total load within the cell: this helps softening the variation among the materials within each cell, leading to a WIP (work in process) minimization within each cell.

The objective function ($F_{individual}$), which represents each chromosome of the population, is the weighted sum of the functions F1 and F2 (adapted from the work of Gonçalves Filho *et. al,.* 2004).

$$F_{individual} = C_1.F_1 + C_2.F_2 \tag{1}$$

2.3.1 Restrictions in the Formation of the Initial Population and the Objective Function

In this work, there are restrictions to the generation of the initial population that must be satisfied. In other words, it should be assured for each individual (chromosome) generated that there are not identical machines in the same cell. Moreover each cell consists of at least two machines. Figure 7 illustrates the main requirements for the formation of the initial population, where (M) stands for the available machines and (K) stands for the number of cells to be formed.

ISSN 2176-5480

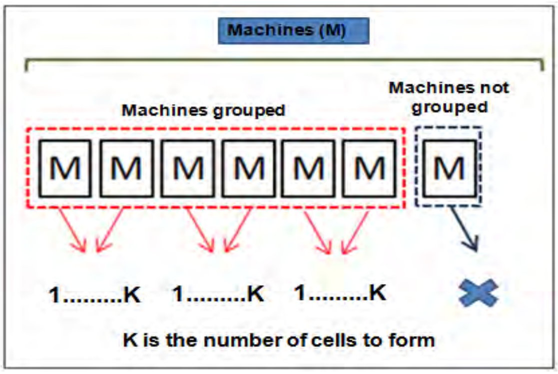

Figure 7. Requirements for the formation of the initial population

The implementation of these constraints in the Matlab toolbox is performed penalizing individuals who do not meet these constraints. This means that every generated individual (chromosome) is checked in relation to the presence of repeated machines or the existence of a single machine in a cell. So when an individual is identified with any of these characteristics, the algorithm penalizes, multiplying the value of the objective function of this element for a very high value, in order to discard this element in future generations.

2.3.2 Scaling of the Objective Function and Selection Criteria

One of the tools available in the Matlab toolbox is the criterion of scheduling of the objective function (*scaling fitness*) in order to rank individuals, according to the objective function value obtained. The criterion used in this work is known as linear order (*Rank*), where individuals are arranged in descending order based on the values of the objective function. Figure 8 illustrates the principle of operation of this tool.

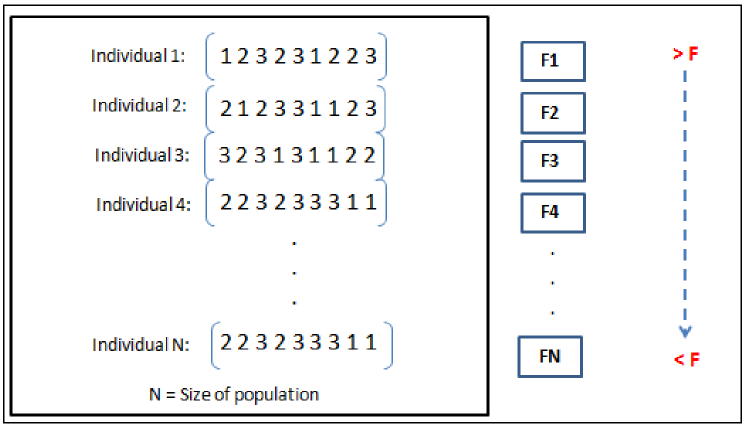

Figure 8. Operational principle of ranking individuals.

After performing the scaling of the objective function, the selection process occurs. The method used in this selection process was Roulette (*Roulette Wheel*), which is applied to problems of GA.

2.4 Genetic Operators Applied to the Problem

The genetic operators are used when the selection of the fittest individuals for breeding is completed. That is, the algorithm sequentially generates a new population for the next generation, where the elements are selected applying genetic operators of recombination (Crossover) and mutation. The following operators were used:

- Recombination (Crossover): *single point* Crossover with a predefined rate of Crossover.
- Mutation: *Uniform* mutation, with a predefine rate of mutation.

2.5 Stopping Criteria and Convergence

The way the algorithm finishes the optimization is based on reaching initially established stopping criteria. The GA toolbox uses the combination of five criteria that can be defined by the user, which are: 1) Generations Limit Value; 2) Time limit; 3) of Fitness Function Limit Value; 4) Number of generations in which the optimal value remains stable

ISSN 2176-5480

(*stall generations*) and 5) Time limit for stabilizing the optimal value between generations (*stall time limit*). In this work, it was considered only the limit of generations (*generations*).

2.6 Functional Characteristics of the Cell Formation Algorithm

The operation of the algorithm can be explained similarly to the basic operation of the GA in the Matlab toolbox. However, there were changes in the subroutines, where specific Matlab scripts were generated to perform certain tasks in the algorithm. In Figure 9 it is explained how the algorithm behaves.

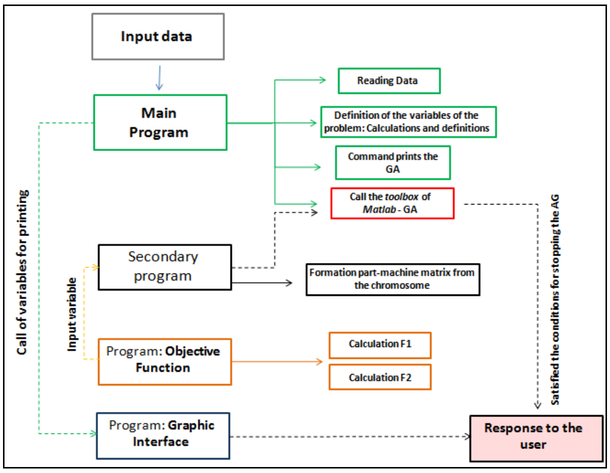

Figure 9. Principle of operation of the proposed algorithm

The algorithm starts assuming it is provided the necessary data for the GA analysis by text file. Figure 10 shows these variables.

These input data are stored in the main program, and based on this input information the respective variables for the problem are set up. Another function of this program is that it performs the initialization of the Matlab toolbox. In order to evaluate the cost function to be optimized the GA subroutine uses two other subroutines, which generate the calculations of the respective objective function and matrix representation in the form of rows and columns rearranged (diagonalized). Finally, after a series of generations, the Matlab toolbox provides the user with a graphical interface.

2.6.1 Graphic Interface Operation

As already emphasized, this paper aims to provide a functional interface so you can interact with the AG, in order to form manufacturing cells. Thus, we created a *.exe, whose goal is to import a data file from the *. dat so they can analyze the best solution according to the analyzed problem. In this interface, information is entered initially, such as the parameters of the GA and the basic requirements for the formation of cells. Figure 10 illustrates such graphical interface that contains the variables of the problem, and can be edited in the system.

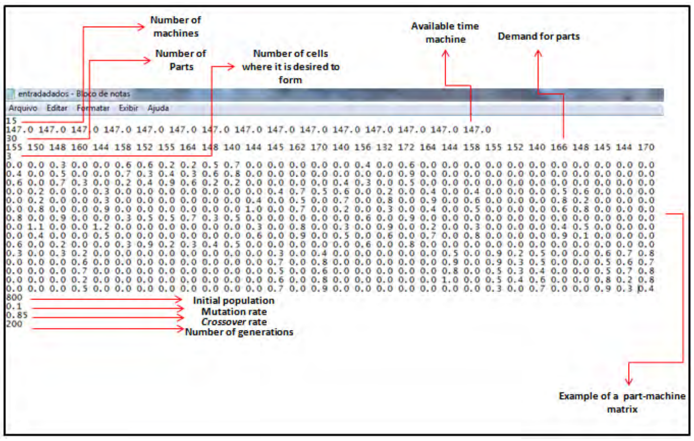

Figure 10. Variable definition interface.

Once defined the variables required to solve the problem, the next step is to include this data in the main program folder (main application interface) to start the simulation. Figure 11 illustrates the initial main interface of the deployed application.

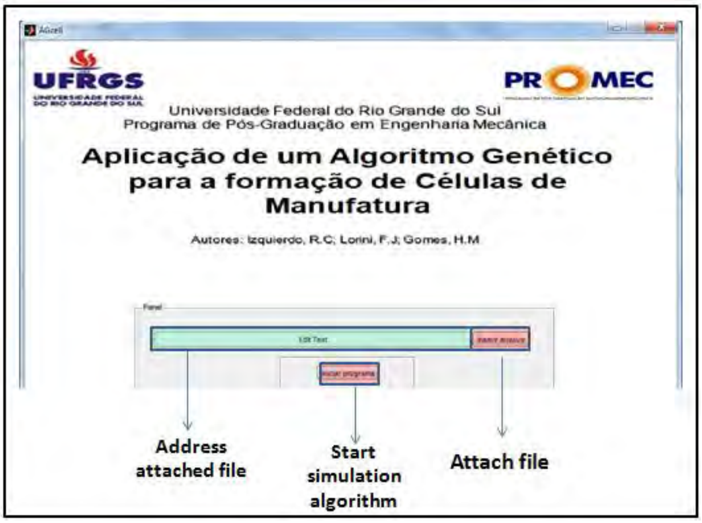

Figure 11. Initial main interface.

3. RESULTS

To validate the proposed application in this paper three examples of optimized cell formation matrices were used, implemented by deterministic and heuristic methods in the literature, in order to compare the accuracy of the application.

3.1 Analysis of Matrix A

This analysis comes from the method proposed by Gonçalves Filho *et al.* (2004), where the problem is initially defined by a matrix composed of 15 to 30 machine parts, so that each element of the matrix corresponds to its manufacturing time. The parameters considered for the implementation of this problem are illustrated in Tables 2 and 3.

Table 2. Parameters of GA for system optimization

Parameters GA	
Initial Population	800
Mutation Rate	0.1
Crossover Rate	0.85
Number of generations	200

Table 3. Parameters for the formation of manufacturing cells

Parameters for cell formation	
Number of machines	15
Number of parts	30
Number of cells	3
Incidence matrix part-machine	Defined in the Gonçalves Filho *et al.*, (2004)
Demand of parts	Defined in the Gonçalves Filho *et al.*, 2004
Available time machine	147 hours, considering a week to i machines

ISSN 2176-5480

The following objective function values were obtained: F1=918 and F2=8.92. Regarding the obtained results it is clear that the values are close to the values in the method analyzed. Regarding the randomness it is observed that the standard deviation obtained after 15 restarts was 416.33. If compared with the magnitude of the cost function, this is relatively high. However, it is important to consider that the objective function value was different from the reference values only in the 2nd and 6th startup, where the values almost doubled the expected result. The representation of the layout obtained by the application is similar to the response obtained in the work of Gonçalves Filho *et al.*(2004).

3.2 Analysis of Matrix B

This analysis comes from the method proposed by Leal *et al.* (2000), where initially the problem is defined using a matrix consisting of 9 and 12 machine parts, so that each element of the array consists of the similarities between the respective part and machine. Similarly from Analysis of Matrix A, Tables 4 and 5 shows the parameters used in the analysis.

Table 4. Parameters of GA for system optimization

Parameters GA	
Initial Population	2000
Mutation Rate	0.1
Crossover Rate	0.9
Number of generations	500

Table 5. Parameters for the formation of manufacturing cells

Parameters for cell formation	
Number of machines	9
Number of parts	12
Number of cells	3
Incidence matrix part-machine	Defined in Leal *et al.* (2000)
Demand of parts	150 parts, considering *i* machine
Available time machine	147 hours, considering a week to i machines

It is observed from the analysis of the results that there are small differences between the responses obtained in the work of Leal *et al.* (2000), and the proposed application, like the arrangement of the groups of machines. That is, the machine 7 is not in the same cell that was presented in the considered example so the family of parts ends up being affected. However, the obtained result by the proposed algorithm is similar to the example considered, since there are cells which have the same characteristics. In this case the differences are acceptable from a productive point of view, in other words, the difference does not represent overloading in one of three cells generated by the application.

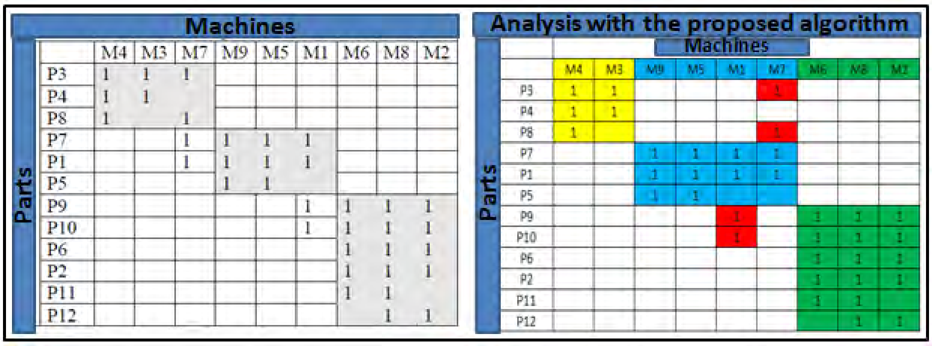

Figure 12. Comparison of the response obtained in the work analyzed and proposed application

Regarding the reliability of the results, it cannot be said that the analysis of this matrix has good accuracy, because the method used for the optimization of the problem was the ROC method, which does not provide the value of the objective function. However, the final cost function result obtained by the proposed algorithm was 660.73, which consists of a relatively low value compared to the other samples analyzed. Furthermore, the results show that after 15 restarts the obtained average value from the objective function was 660.73, which represents a degree of repeatability of the algorithm for implementing this example. With respect to randomness it is observed that the standard deviation obtained after 15 random restarts was equal to zero, this means that the obtained results repeated along all the 15 restarts.

3.3 Analysis of Matrix C

Another analysis addressed in this paper is an application of Burbridge, consisting of a classical deterministic clustering method, presented in the work of Chan *et al.* (1982), where the problem is initially defined by a matrix consisting of 16 to 43 machines parts, so that each array element also consists of similarity between parts and machine. Similarly to matrices A and B, Tables 5 and 6 show the parameters used in the analysis.

Table 5. Parameters of GA for system optimization

Parameters GA	
Initial Population	6000
Mutation Rate	0.1
Crossover Rate	0.9
Number of generations	500

Table 6. Parameters for the formation of manufacturing cells

Parameters for cell formation	
Number of machines	16
Number of parts	12
Number of cells	5
Incidence matrix part-machine	Defined in Chan *et al.* (1982)
Demand of parts	150 parts, considering *i* machine
Available time machine	147 hours, considering a week to *i* machines

The differences between the results obtained by Chan *et al.*(1982) and the proposed application are related to the arrangement of machine groups and corresponding part families respectively. This can be observed in the case of the machine 11 that does not belong to the same cell that was presented on the example of Burbridge. Moreover, it is noted that machines 6 and 8 meet a variety of parts too high, causing a given set of parts not related to any cells formed. The figure13 illustrates the diagonalized Burbridge matrix by of the proposed application.

	AG 1	1	13	39	25	12	31	26	42	37	2	32	38	10	40	28	18	4	17	7	35	6	34	19	23	14	43	5	9	21	41	15	29	8	33	16	27	24	3	20	30	11	22	36
MACHINE	M7	1	1		1																																							
	M10	1	1	1	1	1	1	1																																				
	M1								1	1																																		
	M2								1	1	1	1	1	1	1	1																												
	M9								1	1	1	1	1	1	1	1	1	1																										
	M16								1	1	1	1	1	1			1																											
	M3																		1	1	1		1																					
	M14																		1		1	1																						
	M4																							1	1	1	1							1	1									
	M5																							1	1	1		1	1	1			1											
	M6	1	1	1		1			1	1	1	1			1				1	1		1	1	1	1	1	1	1		1	1				1									
	M8	1				1	1			1	1		1			1								1	1		1		1	1	1	1		1										
	M11																																			1	1	1	1	1				
	M15																							1	1		1		1	1	1	1		1										
	M12																																				1	1			1	1	1	
	M13																																				1	1						1

Figure 13. Results obtained with the proposed application

In the same way of the analysis of matrix B in the work of Chan *et al.* (1982) it cannot be said that the proposed algorithm has good accuracy, because the method used for the optimization of the problem was the DCA method, which also does not provide the value of the objective function. The obtained result of the cost function by the proposed

ISSN 2176-5480

algorithm was 3666.38, which consists of a relatively high value, due to the high number of parts and machinery provided in the matrix proposed by Burbridge. The results show that after 15 random restarts the average obtained from the objective function was 3855.26, which is a relatively close to the value of the minimized objective function. With respect to randomness it is observed that the standard deviation obtained after 15 random restarts was 177.16. If compared with the magnitude of the function obtained it is relatively low. However, it is important to consider that the best response obtained related to the optimization algorithm (GA) does not match the best response in terms of layout, i.e., the optimized objective function does not give the best visual layout. Thus, it was adopted in response to layout representation Burbridge, whose objective function value is 4010.5. Here it is important to highlight that this results shows that the used cost function in the optimization is missing some especial features that could penalize configurations such as those obtained by the GA. In fact, the obtained configuration of cell formation presents a cost function that is smaller than that of the Burbridge, however this cost function does not account for off-cell parts, which were not included in the cost function. Some further research is necessary in improving this cost function.

4. CONCLUSIONS

Considering that a model of cell formation layout is most useful when we analyze the work in a manufacturing environment in which production is made in small batches with a high diversity of products, it is expected that this work will provide the final user with an application to assist in the grouping of machines into cells and parts into their respective families. This will provide another tool for analysis, which is an important step in layout design.

Among the various methods that are used in layout planning, heuristic methods have been widely studied. In this context, the present study implemented a heuristic classic GA in a conventional tool such as Matlab. This methodology is valid for both design of new systems and cases where one wants an optimization of the existing layout, depending on a given market demand, or through a preliminary analysis of the production flow of a manufacturing environment, aiming at a rational planning of the layout.

Considering the implementation of an optimization algorithm, using the GA method, we can infer the following considerations:

- In all examples shown it can be seen that the efficiency of the GA is directly related parameters arbitrated, as the population size, rate of mutation, Crossover rate and number of generations.
- In the analysis of the matrices B and C, it was not possible to measure the accuracy of the methods analyzed because they were not implemented with Genetic Algorithms or other heuristic method, that is, there is a value of objective function reference to conduct a comparative analysis. However, these examples showed that they both have small randomness, with respect to the magnitude of the objective function analyzed, indicating the relative robustness in the results.
- In the last example, it is important to highlight that the results showed that the used cost function in the optimization was missing some especial features that could penalize configurations such as those obtained by the GA. In fact, the obtained configuration of cell formation in this last example presented a cost function that was smaller than that obtained with Burbridge method, however this cost function does not account for parts not belonging to cells of machines, which were not included in the cost function. Some further research is necessary to improve this cost function in order to take into account such features.

Analyzing the obtained results in this work, the method implemented using the Genetic Algorithm is presented as an alternative to the cell formation.

5. REFERENCES

Arkat, J; Abdollahzadeh, H; Ghahve, H. A new branch and bound algorithm for cell formation problem. Applied Mathematical Modelling 36, p 5091-5100, 2012.

Arzi ,Y; Bukchin, J; Masin, M. An efficiency frontier approach for the design of cellular manufacturing systems in a lumpy demand environment. European Journal of Operational, p 346-364, 2001.

Chan, H.M.; Milner, D.A. Direct clustering algorithm for group formation in cellular manufacture. Journal of manufacturing system, v. 1, ed. 1, p 65-75, 1982.

Corrêa, H.L.; Corrêa, C.A. Administração de produção e de operações: manufatura e serviços: uma abordagem estratégica. 2.ed. Atlas, São Paulo, 2006.

Filho, E.V.G; Junior, J.H.C.G; Argoud, A.R.T.T. Algoritmo Genético para formação de células de fabricação. ENEGEP, Florianópolis, SC, 2004.

Groover, M.P. Automação industrial e sistemas de manufatura/ Mikell Groover; tradução Jorge Ritter, Luciana do Amaral Teixeira, Marcos Vieira; revisão técnica José Hamilton Chaves Gorgulho Júnior - 3. ed. - São Paulo: Pearson Prentice Hall, 2011.

Leal, R. A. ; Correia Filho, A. N.; Maestrelli, N. C.; Batocchio, A. . Uma aplicação do algoritmo ROC (Rank Order Clustering) em manufatura celular. In: Congresso Nacional de Engenharia Mecânica, 2000, NATAL, RN. Anais do Congresso Nacional de Engenharia Mecânica - CONEM 2000.

Lorenzatto, J.T; Ribeiro, J.L.D. Projeto de Layout alinhado às praticas de produção enxuta em uma empresa siderúrgica de grande porte. ENEGEP, Foz do Iguaçu, PR, 2007.

Lorini, F.J. Tecnologia de grupo e organização da manufatura. Editora UFSC, Florianópolis, 1993.

Sayadi, M.K; Ashkan, H. Naini, S.G.J. Firefly- inspired algorithm for discrete optimization problems: An application to manufacturing cell formation. Journal of Manufacturing Systems, p 78-84, 2013.

Singh, N.; Rajamani, D. Cellular Manufacturing Systems: Design, Planning and Control. Chapman & Hall, London, 1996.

Slack, N. Brandão, A.B. Corrêa, H. Administração da produção edição compacta. Atlas, São Paulo, 1999.

Tahara, C.S. Carvalho, M.M, Filho, E.V.G. Revisão das Técnicas para Formação de Células de Manufatura. ENEGEP, 1997.

Vernugopal, V. Narendran, T.T. A genetic algorithm approach to the machine-component grouping problem with multiple objectives. Computers in Industrial Engineering. Vol 22, n.4.p.469-480.

6. RESPONSIBILITY NOTICE

The author(s) is (are) the only responsible for the printed material included in this paper.