Solving Manufacturing Cell Design Problems using an Artificial Fish Swarm Algorithm

Ricardo Soto[1,2,3], Broderick Crawford[1,4,5], Emanuel Vega[1], and Fernando Paredes[6]

[1] Pontificia Universidad Católica de Valparaíso, Chile
[2] Universidad Autónoma de Chile, Chile
[3] Universidad Cientifica del Sur, Lima, Peru
[4] Universidad Central de Chile, Chile
[5] Universidad San Sebastián, Chile
[6] Escuela de Ingeniería Industrial, Universidad Diego Portales, Chile
{ricardo.soto,broderick.crawford}@ucv.cl
emanuelvm88@gmail.com
fernando.paredes@udp.cl

Abstract. The design of manufacturing cells is a manufacturing strategy that involves the creation of an optimal design of production plants, whose main objective is to minimize movements and exchange of material between these cells. Optimal solution of large scale manufacturing cell design problems (MCDPs) are often computationally unfeasible and only heuristic and approximate methods are able to handle such problems. Artificial fish swarm algorithm (AFSA) belongs to the swarm intelligence algorithms, which based on population search, are able to solve complex optimization problems. In this paper we present an AFSA-based approach to solve the MCDP by using the classic Boctor's mathematical model. The obtained results show that the proposed algorithm produces optimal solutions for all the 50 studied instances.

Keywords: Manufacturing Cell Design Problem, Artificial Fish Swarm Algorithm, Metaheuristic.

1 Introduction

The design of manufacturing cells has emerged in the last two decades as innovation for manufacturing strategy, this strategy involves the creation of an optimal design of production plants. The manufacturing cell design problem (MCDP) considers grouping similar parts into part-families. Ideally, each of these families is processed by a dedicated cluster of manufacturing facilities called manufacturing cell, where the main goal is to minimize movement and exchange of material between cells. In this context, the cell formation problem has been matter of considerable research, where Burbidge with his production flow analysis in 1963, becomes one of the first to propose a process to solve this concern.

In this paper, we propose an Artificial Fish Swarm Algorithm (AFSA) to solve the MCDP. The AFSA was presented by X. L. Li in 2002 [1], it is a technique based on swarm behaviors and was inspired from social conduct of fish swarm in nature. AFSA, works based on population, random search, and behaviorism. This algorithm has been used in optimization applications, such as clustering [2,3], machine learning [4,5], PID control [6], data mining [7], and image segmentation [8]. We illustrate promising results where the proposed approach noticeable competes with previous reported techniques for solving manufacturing cell design problems.

The rest of this paper is organized as follows. In Section 2, we present the related work. Section 3 describes the mathematical model of the MCDP. The AFSA is explained in Section 4. Finally, Section 5 illustrates the experimental results, followed by conclusions and future work.

2 Related Work

The problem of formulating cells has been the subject of considerable research, where Burbidge, with his production flow analysis in 1963, becomes one of the first to solve this problem [9]. Other approaches try to solve the MCDP by attempting to determine the part families, finding only partial solutions [10,11]. Most of these methods are based on the incidence machine-part matrix, and can be divided into hierarchical and non-hierarchical clustering. For instance Shargal [12] presented search algorithms and clustering efficiency measures for machine-part matrix, Seifoddini and Hsu [13] work with clustering algorithms in cellular manufacturing, and Srinivasan [14] use clustering algorithm for machine cell formation in group technology using minimum spanning tree. Also, graph theoretical mathematical programming methods, for instance Deutsch [15] uses an improved p-median model for cell formation, Atmani [16] presents a mathematical programming approach to joint cell formation and operation allocation in cellular manufacturing, Adil [17] propose a mathematical model for cell formation considering investment and operational costs, Purcheck [19] presents a linear-programming method for the combinatorial grouping of an incomplete set, Olivia Lopez and Purcheck [20] works in a load balancing for group technology planning and control, and Boctor [[22], [11], [21]] presented work with cell formation. The implementation of approximate methods, such as metaheuristics, has been material of work for researchers devoted to solve cell formation problems.Durn, Rodriguez and Consalter [23] combines particle swarm optimization and discrete position update scheme techniques for manufacturing cell design. Wu, Chang, and Chung [24] present a simulated annealing (SA) approach, and Venugopal and Narendran [25] propose the use of genetic algorithms (GA), which would be used later by Gupta, Gupta, Kumar, and Sundaram [26] but focused in multi-objective optimization approach. In this paper, our goal is to employ a modern metaheuristic to report better solutions than the ones applied before to manufacturing cell design.

3 Manufacturing Cell Design Problem

Manufacturing strategy consisting in creating an optimal design of production plants, which are composed of manufacturing cells and machines that process subsets of parts forming families, determined according to the similarity of them. The objective is to minimize movement and exchange of material between cells, in order to reduce production costs and increase productivity. We represent the processing requirements of machine parts through an incidence matrix called machine-part. This matrix contains a binary domains and is denoted as A, where a_{ij}=1 means that machine i is necessary to process part j and a_{ij}=0 otherwise. A rigorous mathematical formulation of machine-part grouping problem with these objectives is given by Boctor [22] and its as follows:

- let M, the number of machines,
- let P, the number of parts,
- let C, the number of cells,
- let i, the index of machines (i = 1,...,M),
- let j, the index of parts (j = 1,...,P),
- let k, the index of cells (k = 1,...,C),
- $A = [a_{ij}]$ the binary machine-part incidence matrix MxP,
- M_{max}, the maximum number of machines per cell. We selected as the objective function to be minimized the number of times that a given part must be processed by a machine that does not belong to the cell that the part has been assigned to. Let:

$$y_{ik} = \begin{cases} 1 & \texttt{if machine i} \in \texttt{cell k;} \\ 0 & \texttt{otherwise;} \end{cases}$$

$$z_{jk} = \begin{cases} 1 & \texttt{if part j} \in \texttt{family k;} \\ 0 & \texttt{otherwise;} \end{cases}$$

The problem is represented by the following mathematical model:

$$Minimize \sum_{k=1}^{C} \sum_{i=1}^{M} \sum_{j=1}^{P} a_{i_j} z_{j_k} (1 - y_{i_k})$$

Subject to

$$\sum_{k=1}^{C} y_{ik} = 1 \qquad \forall i$$

$$\sum_{k=1}^{C} z_{jk} = 1 \qquad \forall j$$

$$\sum_{i=1}^{M} y_{ik} \leq M_{max} \qquad \forall k$$

4 Artificial Fish Swarm Algorithm

AFSA is an ordered and finite set of bionic operations for optimization, based on the study of intelligent behavior of the fish swarm. This means that, in an area of water, the fish can often find places that contain many nutrients by himself or following other fish. Therefore, where there is the largest number of fish is usually the place that has the most nutrients [27,28].

4.1 Proposed Algorithm

AFSA simulates the behavior of a fish swarm, which shapes the Artificial Fish (AF) that seeks an optimal solution in the solution space. The AF perceives external concepts with sense of sight. Current position of AF is shown by vector X=(X_1, X_2, ... , X_n). The visual is equal to sight field of AF and X_v is a position in visual where the AF wants to go. Then if X_v has better food consistence than current position of AF, it goes one step toward X_v which causes change in AF position from X to X_{next}, but if the current position of AF is better than X_v, it continues searching in his visual area. Food consistence in position X is the fitness value of the current position and it is shown with f(x). The step is equal to maximum length of the movement. The AF consists in two parts, variables and functions. Variables include X (current AF position), step (maximum length step), visual (sight field), *trynumber* (the maximum test interactions and tries) and crowd factor σ (0< σ <1). Also his functions consist on the prey behavior, free move behavior, swarm behavior and follow behavior. In each step for the optimization process the AF search for locations with better fitness values all over the search space, the AF is behaviors are explained:

1. Prey behavior: This behavior is an individual behavior that each AF performs independently and performs a local search around itself. Every AF by performing this behavior attempts try-number times to move to a new position with better fitness.
2. Follow behavior: The best AF of the swarm locates the best found position so far by the swarm. In follow behavior, each of AF moves one step toward the best AF of swarm.
3. Swarm behavior: To ensure the security and integrity of each AF, they trend to group up. this, they follow 3 basis rules, separation, alignment and cohesion.
4. Free Move behavior: This is a special behavior where the AF search for a random position, if it is not a better one, the AF will go through a modification of the new step to ensure not to be stagated in a local optimum, otherwise the AF will move ahead.

Performing these four behaviors based on an algorithm procedure [29] it is described as follows:

1. Initialization;
2. Calculate the fitness value;
3. For each AFi, where (i=1, 2, .. , N);

   3.1 Follow behavior; determine whether the state after the follow is better than the previous state, if so, advances to step 4, otherwise we resort to step 3.2;
   3.2 Prey behavior; determine whether the search state is better than the old one *Trynumber* times, if so proceeds to step 4, otherwise we resort to Step 3.3;
   3.3 Swarm behavior; determine whether the state after grouping is better than previous state, if so, advances to step 4, otherwise we resort to step 3.4 ;
   3.4 Movement;

4. Refresh the current best value;
5. If it concludes the number of Iterations, out; otherwise, return to step 3;

5 Experimental result

The effectiveness of our proposed approach has been tested using the incidence matrices [22]. In this paper these 10 problems were solved using the model presented in section 2 with different sets of parameters. For the experimental evaluation, the parameters employed are defined as follows: 16 Machines, 30 Parts, and a combination between 2 Cells with 8, 9, 10, 11, 12 Mmax. Concerning the AFSA, the configuration uses 40 as initial population size, a crowd factor σ between 0.9 and 0.5 and *Trynumber* 4 and 50.000 iterations for the AFSA. The algorithm has been implemented using Java and launched on a 2.4 GHz Intel Core i7 with 8GB RAM running Windows 8. Table 1 and table 2 show detailed information of the results obtained by our approach. Here, we compare our results with the ones reported in [22] and [23]. Concerning to the others values, represented as column OPT, is the Boctors values, column PSO, the best value obtained by Particle Swarm Optimization in[23], column SA, the best value obtained by Simulated Annealing reported in [22] and column AFSA, the best value obtained by our proposed AFSA. Table 3 shows the relative percentage derivation (RPD) and the average values of our AFSA in order to evaluate the quality of every solution.

Table 1: Results of AFSA using 2 cells

Pblm	Mmax = 8				Mmax = 9				Mmax = 10			
	Opt	SA	PSO	AFSA	Opt	SA	PSO	AFSA	Opt	SA	PSO	AFSA
1	11	11	11	11	11	11	11	11	11	11	11	11
2	7	7	7	7	6	6	6	6	4	10	5	4
3	4	5	5	4	4	4	4	4	4	4	5	4
4	14	14	15	14	13	13	13	13	13	13	13	13
5	9	9	10	9	6	6	8	6	6	6	6	6
6	5	5	5	5	3	3	3	3	3	5	3	3
7	7	7	7	7	4	4	5	4	4	4	5	4
8	13	13	14	13	10	20	11	10	8	15	10	8
9	8	13	9	8	8	8	8	8	8	8	8	8
10	8	8	9	8	5	5	8	5	5	5	7	5

Table 2: Results of AFSA using 2 cells

Pblm	Mmax = 11				Mmax = 12			
	Opt	SA	PSO	AFSA	Opt	SA	PSO	AFSA
1	11	11	11	11	11	11	11	11
2	3	4	4	3	3	3	4	3
3	3	4	4	3	1	4	3	1
4	13	13	13	13	13	13	13	13
5	5	7	5	5	4	4	5	4
6	3	3	4	3	2	3	4	2
7	4	4	5	4	4	4	5	4
8	5	11	6	5	5	7	6	5
9	5	8	5	5	5	8	8	5
10	5	5	7	5	5	5	6	5

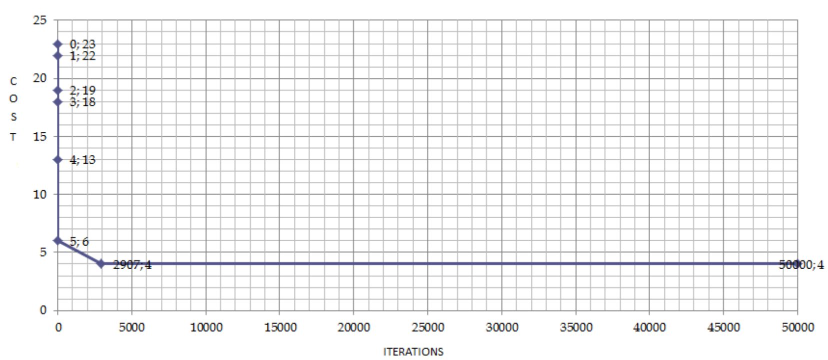

Fig. 1: Graph of Problem 7 solved by AFSA with Mmax 10

According to table 1 and table 2, results obtained by AFSA are better than or equal to those reported result by PSO and SA in all the test problems from the literature. To be more specific, the proposed AFSA has high quality solutions and good performance. Thus, shown in Fig. 1, we ilustrate the convergence rate by AFSA accomplishing robust search capability in early stages. Also, it is reported by the RPD value the quality, which quantifies the deviation of the objective value that our approach has a high consistency on it is solutions.

Table 3: Average and relative percentage derivation

	Mmax=8		Mmax=9		Mmax=10		Mmax=11		Mmax=12	
Pblm	Avg	RPD(%)	Avg	RPD(%)	Avg	RPD(%)	Avg	RPD(%)	Avg	RPD(%)
1	11	0	11	0	11	0	11	0	11	0
2	7	0	6	0	4	0	3	0	3	0
3	4	0	4	0	4	0	3	0	1	0
4	14	0	13	0	13	0	13	0	13	0
5	9	0	6	0	6	0	5	0	4	0
6	5	0	3	0	3	0	3	0	2	0
7	7	0	4	0	4	0	4	0	4	0
8	13	0	10	0	8	0	5	0	5	0
9	8	0	8	0	8	0	5	0	5	0
10	8	0	5	0	5	0	5	0	5	0

6 Conclusion and future work

The metaheuristic usually applies to problems that have no specific algorithm or heuristic that gives a satisfactory solution. Thus, we employed the search algorithm AFSA to tackle the design of manufacturing cells, where the execution of the natural behaviors, minimizes the movement and exchange of material between cells; looking for an optimization of time and costs. The convergence rate demonstrated by AFSA, it is related to the values of its parameters. Initially high values are used, such as a crowd factor σ of 0.8 or a *Trynumber* of 20. After analyzing the results it is concluded that the use of parameters with smaller or dynamic values improve the performance. Also, parameters as crowd factor, *Trynumber*, and distance calculation were tuned from his original values, standard AFSA, and example is the use of Hamming distance. We also applied fix to possible solutions that could not satisfy the constraint and fixed the swarm behavior to be more compatible with the nature of the problem ending with better results and convergence rate.

Also, there are some relevant works to pursue in the future. First, change how we tackle the MCDP, there is some heavy initial work done to set good initialization, and for every time we generate the matrix part-cell. Second, work on a better adaptation of the swarm behavior, this will benefit also to improve the efficiency of the algorithm.

7 Acknowledgements

Ricardo Soto is supported by Grant CONICYT/FONDECYT/INICIACION/ 11130459, Broderick Crawford is supported by Grant CONICYT/FONDECYT/ 1140897, and Fernando Paredes is supported by Grant CONICYT/FONDECYT/ 1130455.

References

1. Li, L.X., Shao, Z.J., Qian, J.X. *An Optimizing Method Based on Autonomous Animate: Fish Swarm Algorithm.* In: Proceeding of System Engineering Theory and Practice, pp. 32–38. (2002).
2. Hi, S., Belacel, N., Hamam, H., Bouslimani, Y. *Fuzzy Clustering with Improved Artificial Fish Swarm Algorithm.* In: International Joint Conference on Computational Sciences and Optimization 09, pp. 317–321, Hainan (2009).
3. Xiao, L. *A Clustering Algorithm Based on Artificial Fish Swarm.* In: 2nd International Conference on Computer Engineering and Technology, pp. 766–769, Chengdu (2010).
4. Yazdani, D., Golyari, S., Meybodi, M.R. *A New Hybrid Algorithm for Optimization Based on Artificial Fish Swarm Algorithm and Cellular Learning Automata.* In: 5 International Symposium on Telecommunication (IST), pp. 932–937, Tehran (2010).
5. Yazdani, D., Nadjaran Toosi, A., Meybodi, M.R. *Fuzzy Adaptive Artificial Fish Swarm Algorithm.* In: 23th Australian Conference on Artificial Intelligent, Adelaide (2010).
6. Luo, Y., Zhang, J., Li, X. *The Optimization of PID Controller Parameters Based on Artificial Fish Swarm Algorithm.* In: IEEE International Conference on Automation and Logistics, pp. 1058–1062, Jinan (2007)
7. Zhang, M., Shao, C., Li, M., Sun, J. *Mining Classification Rule with Artificial Fish Swarm.* In: 6 World Congress on Intelligent Control and Automation, pp. 5877–5881, Dalian (2006).
8. Li, C.X., Ying, Z., JunTao, S., Qing, S.J. *Method of Image Segmentation Based on Fuzzy C-means Clustering Algorithm and Artificial Fish Swarm Algorithm.* In: International Conference on Intelligent Computing and Integrated Systems (ICISS), Guilin (2010).
9. Xambre, A.R., and Vilarinho, P.M., *A simulated annealing approach for manufacturing cell formation with multiple identical machines*, European Journal of Operational Research 151, pp. 434446, 2003.
10. Kusiak, A., *The part families problem in Flexible Manufacturing Systems*, Annals of Operational Research 3, 279300, 1985.
11. Xambre, A. R. and Vilarinho, P. M., *A simulated annealing approach for manufacturing cell formation with multiple identical machines*, European Journal of Operational Research 151, pp. 434446, 2003.

12. Shargal, M., Shekhar, S., Irani, S.A., *Evaluation of search algorithms and clustering efficiency measures for machine-part matrix clustering*, IIE Transactions 27 (1), 4359, 1995.
13. Seifoddini, H., Hsu, C.-P., *Comparative study of similarity coefficients and clustering algorithms in cellular manufacturing*, Journal of Manufacturing Systems 13 (2), 119127, 1994.
14. Srinivasan, G., *A clustering algorithm for machine cell formation in group technology using minimum spanning tree*, International Journal of Production Research 32 (9), 21492158, 1994.
15. Deutsch, S.J., Freeman, S.F., Helander, M., *Manufacturing cell formation using an improved p-median model*, Computers and Industrial Engineering 34 (1), 135146, 1998.
16. Atmani, A., Lashkari, R.S., Caron, R.J., *A mathematical programming approach to joint cell formation and operation allocation in cellular manufacturing*, International Journal of Production Research 33 (1), 115, 1995.
17. Adil, G.K., Rajamani, D., Strong, D., *A mathematical model for cell formation considering investment and operational costs*, European Journal of Operational Research 69 (3), 330341, 1993.
18. Kusiak, A., and Chow, W., *Efficient solving of the group technology problem*, Journal of Manufacturing Systems, 6, 117124, 1987.
19. Purcheck, G., *A linear-programming method for the combinatorial grouping of an incomplete set*, Journal of Cybernetics, 5, 5158, 1975.
20. Olivia-Lopez, E., and Purcheck, G., *Load balancing for group technology planning and control*, International Journal of MTDR, 19, 259268, 1979.
21. Soto,R., Kjellerstrand, H., Durn, O., Crawford, B., Monfroy, E. and Paredes,F., *Cell formation in group technology using constraint programming and Boolean satisfiability*, Expert Systems with Applications 39, pp. 11423-11427, 2012.
22. Boctor, F. F., *A linear formulation of the machine-part cell formation problem*, International Journal of Production Research, Volume 29, Nro 2, pp. 343-356, 1991.
23. Durn, O., Rodriguez, N., Consalter, L., *Collaborative particle swarm optimization with a data mining technique for manufacturing cell design.* Expert Systems with Applications, 37(2), pp. 15631567, 2010.
24. Wu, T., Chang, C., Chung, S. *A simulated annealing algorithm for manufacturing cell formation problems.* Expert Systems with Applications, 34(3), pp. 16091617, 2008.
25. Venugopal, V., Narendran, T. T. *A genetic algorithm approach to the machine-component grouping problem with multiple objectives.* Computers and Industrial Engineering, 22(4), pp. 469480, 1992.
26. Gupta, Y., Gupta, M., Kumar, A., Sundaram, C. *A genetic algorithm-based approach to cell composition and layout design problems.* International Journal of Production Research, 34(2), pp.447482, 1996.
27. Yazdani, D., Golyari, S. and Reza, M. M., *A New Hybrid Approach for Data Clustering*, 5th International Symposium on Telecommunication (IST), Tehran, pp. 932-937, 2010.
28. WANG, L. and MA, L., *A Hybrid Artificial Fish Swarm Algorithm for Bin-packing Problem*, International Conference on Electronic and Mechanical Engineering and Information Technology, pp.27-29, 2011.
29. Zhang, M., et al., "*Mining Classification Rule with Artificial Fish Swarm*", 2006, pp. 5877-5881.