Expert Systems with Applications 37 (2010) 5476–5483

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

A numerical comparison between simulated annealing and evolutionary approaches to the cell formation problem

Andres Pailla [a], Athila R. Trindade [b], Victor Parada [a,*], Luiz S. Ochi [b]

[a] Departamento de Ingeniería Informática, Universidad de Santiago de Chile, Av. Ecuador 3659, Santiago, Chile

[b] Instituto de Computaçao, Universidade Federal de Fluminense, Rua Passo da Pátria 156, Bloco E, 3° andar, São Domingos, Niterói, RJ, CEP 24210-240, Brazil

ARTICLE INFO

Keywords:
Heuristics
Evolutionary algorithms
Simulated annealing
Cell formation problem
Clustering

ABSTRACT

The cell formation problem is a crucial component of a cell production design in a manufacturing system. This problem consists of a set of product parts to be manufactured in a group of machines. The objective is to build manufacturing clusters by associating part families with machine cells, with the aim of minimizing the inter-cellular movements of parts by grouping efficacy measures. We present two approaches to solve the cell formation problem. First, we present an evolutionary algorithm that improves the efficiency of the standard genetic algorithm by considering cooperation with a local search around some of the solutions it visits. Second, we present an approach based on simulated annealing that uses the same representation scheme of a feasible solution. To evaluate the performance of both algorithms, we used a known set of CFP instances. We compared the results of both algorithms with the results of five other algorithms from the literature. In eight out of 36 instances we considered, the evolutionary method outperformed the previous results of other evolutionary algorithms, and in 26 instances it found the same best solutions. On the other hand, simulated annealing not only found the best previously known solutions, but it also found better solutions than existing ones for various problems.

1. Introduction

Production management is an important factor in the success of an industry. Its goal is to organize the production environment in order to save time and production costs without loss of quality. Industries that focus on small variety and high production volume usually organize the production environment into production lines, and each production line consists of various kinds of machines devoted exclusively to the production of a single product.

In the case of an industry with high variety and medium production volume, the machines need not be dedicated only to the production of a single product. Therefore, one kind of production organization comprises the development of clusters of machines with identical functions (for example, groups of lathes or of milling machines), forming specialized departments. In this way, any part of a given product that needs manufacturing operations by more than one kind of machine will need to go through all the groups that contain the kinds of machines required for its complete manufacture. As a result, this kind of organization favors the gathering of a large volume of material that the various groups of machines will handle, increasing the time needed for production and making it more difficult to control and maintain the production system. An organizational model to represent this scheme of work, called cellular manufacture, has been used over the last two decades. In this scheme, machines with different functions are grouped in one cell that is devoted to the production of a family of parts that are very similar to each other.

An example of the formation of manufacturing cells is shown in Fig. 1a and b. Consider a production flow consisting of six parts, four machines and 14 activities (incidence matrix positions with a value of 1). Fig. 1a represents the input matrix, and b represents a possible solution matrix with the formation of two cells (families) identified by the shaded area. The manufacturing cells consist of machines (M2,M3) with family of parts (1,3,6) and (M1,M4) with family of parts (2,4,5).

The clusters of machine cells and parts families must have a high degree of independence between them and must be characterized by the presence of few exceptional elements which correspond to those elements with a value of 1 that do not belong to any cluster. Such elements represent the presence of parts that need more than one machine cell for their complete manufacture, and therefore they generate the need for movement of material between different manufacturing cells. It is also desirable to have only a few voids within the clusters, because this involves differences in the work loads of the machines belonging to the same production cluster. In other words, it is desired that each cluster

* Corresponding author. Tel.: +56 (2) 7180900; fax: +56 (2) 7795892.
E-mail addresses: apailla@diinf.usach.cl (A. Pailla), vparada@diinf.usach.cl, victor.parada@usach.cl (V. Parada), satoru@ic.uff.br (L.S. Ochi).

0957-4174/$ - see front matter
doi:10.1016/j.eswa.2010.02.064

	P1	P2	P3	P4	P5	P6
M1		1			1	
M2	1	1	1		1	1
M3	1		1			1
M4		1		1	1	1

a

	P1	P3	P6	P2	P4	P5
M2	1	1	1	1		1
M3	1	1	1			
M1				1		1
M4			1	1	1	1

b

Fig. 1. Cluster matrix.

should be as dense as possible. In the example of Fig. 1b a void is identified in the second cluster given by the (M1,P4) component.

In practice it is uncommon to find production systems that may become manufacturing systems with completely independent clusters (obviously excluding the particular case in which the system is composed of a single cluster). In view of this, many practitioners have tried to solve the independence of the clusters by doubling the number of machines responsible for the appearance of exceptional elements, or by outsourcing the corresponding activities. However, a decision in this sense must be supported by a technical and economic feasibility analysis, which in some situations puts in evidence the impossibility of choosing such alternative. The solution must then consider the forced optimization of the available resources.

The problem of defining an optimum set of *clusters* that includes parts and machines is known as the cell formation problem (CFP). Considering p clusters, the number of possible ways of grouping n parts and m machines in the p clusters, which in turn corresponds to the number of feasible solutions of the CFP, is given by (1) (Wang, 1999). Considering that this number becomes computationally infeasible, even for small numbers of machines and parts, in recent years research has been more often directed toward heuristic search models

$$\left(\left(\sum_{i=1}^{p}(-1)^{p-i}i^{n}/[i!(p-i)!]\right)*\left(\sum_{i=1}^{p}(-1)^{p-i}i^{m}/[i!(p-i)!]\right)\right) \qquad (1)$$

The problem has been studied with several variations, derived from considering different functions to be optimized and different design approaches (Yin & Yasuda, 2006). Some researchers have studied the CFP jointly with the sequencing problem, while others have considered it jointly with the problem of defining machine layout. Literature reviews about CFP can be found in the papers of Selim, Askin, and Vakharia (1998) and Defersha and Chen (2006).

In previous work, researchers have applied mathematical programming and metaheuristics to the CFP and its variants. They have used evolutionary algorithms: a kind of metaheuristic that emulates the process of natural evolution. Starting from a population of solutions where each solution is evaluated, the crossover, mutation and selection operators applied to the current population give rise to a new population. When each solution of the problem is represented as a binary string, the resulting algorithm is known as genetic (GA) (Goldberg, 1989; Silver, 2004). Joines, Culbreth, and King (1996) developed a GA to solve the CFP with an integer programming formulation with a multicriteria objective function, and previously knowing an upper bound on the number of clusters. Mak, Wong, and Wang (2000) also proposed an adaptive genetic algorithm in which the rate of using mutation and crossover changes dynamically with the efficiency of such operators during the execution of the algorithm. Recently, researchers have argued that it is possible to make genetic algorithms for solving optimization problems perform better by considering a local improvement process over each new individual generated in a population. In this respect, Goncalves and Resende (2004) used a local search algorithm coupled with the GA to improve the CFP's solution in several instances. James, Brown, and Keeling (2007) have explored a similar approach using the same local search algorithm. The local search used in both cases is a two-stage iterative process that first analyzes the assignment of one part to a group of machines, then analyzes the assignment of a machine to a family of parts. To identify the best assignment, the algorithm evaluates the grouping efficacy of each assignment. The results of James et al. (2007) are as good or better than those obtained with a grouping genetic algorithm (GGA), studied by Brown and Sumichrast (2001). Muruganandam, Prabhaharan, Asokan, and Baskaran (2005) have also considered the idea of a local search, has by considering tabu search for the local optimization stage (Glover & Laguna, 1998). In this case, the problem under consideration is a CFP variant that includes machine sequencing and combines an objective function that minimizes the total number of moves along with the cell load variation. Its results in four test instances also show that the algorithm that combines GA with TS performs better than GA and TS in its standard version.

Simulated annealing (SA) is another kind of metaheuristics that have been used extensively to solve combinatorial optimization problems. By simulating the phenomenon that takes place in the cooling of pure substances from the liquid to the solid state, SA improves a solution to an optimization problem gradually until it finds the best solution in the search space (Kirkpatrick, Gelatt, & Vecchi, 1983; Silver, 2004). In each iteration, the algorithm accepts a randomly generated solution in the neighborhood of the current solution directly if it is better or probabilistically if it is worse. Wu, Chang, and Chung (2008) have studied CFP by SA, but in a similar way to the evolutionary approaches, they proposed a local optimization stage around the current solution, similar to the generation of neighboring solutions used by the tabu search method. They use two kinds of moves; single-move and exchange-move, both of which are operations that exhaustively revise the neighborhood of the current solution in order to obtain the best solution. They compare their results with a genetic algorithm that solves a traveling salesman problem directly associated with the CFP (Cheng, Gupta, Lee, & Wong, 1998), improving the grouping efficacy.

In this paper, we present a new approach for both search strategies: evolution coupled with a local search, and simulated annealing. In the first case, we study the impact both of the use of local search, and of a procedure to generate an initial population with a constructive heuristic algorithm. In the second case, considering the same representation scheme, we propose a simulated annealing approach. Section 2 describes both algorithms, and Section 3 presents a computational experiment showing that our algorithm is efficient. Finally, Section 4 presents the main conclusions of the study.