Expert Systems with Applications 37 (2010) 1563–1567

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Collaborative particle swarm optimization with a data mining technique for manufacturing cell design

Orlando Durán [a,*], Nibaldo Rodriguez [b], Luiz Airton Consalter [c]

[a] Escuela de Ing. Mecánica, Pontificia Universidad Católica de Valparaiso, Chile
[b] Escuela de Ingeniería Informática, Pontificia Universidad Católica de Valparaíso, Chile
[c] FEAR Universidade de Passo Fundo, Passo Fundo, RS, Brazil

ARTICLE INFO

Keywords:
Manufacturing cells
Machine grouping
Particle swarm optimization

ABSTRACT

In recent years, different metaheuristic methods have been used to solve clustering problems. This paper addresses the problem of manufacturing cell formation using a modified particle swarm optimization (PSO) algorithm. The main modification that this work made to the original PSO algorithm consists in not using the vector of velocities that the standard PSO algorithm does. The proposed algorithm uses the concept of proportional likelihood with modifications, a technique that is used in data mining applications. Some simulation results are presented and compared with results from literature. The criterion used to group the machines into cells is based on the minimization of intercell movements. The computational results show that the PSO algorithm is able to find the optimal solutions in almost all instances, and its use in machine grouping problems is feasible.

1. Introduction

Cellular manufacturing is an organizational approach based on group technology (GT). Cellular manufacturing aims to divide the plant into a certain number of cells. Each cell contains machines that process similar types or families of products. Manufacturing cells (MC) provide considerable cost and productivity benefits to practical manufacturing environments. Other considerable benefits to be gained by grouping machines into cells can be found in literature (Selim, Askin, & Vakharia, 1998).

The major issue in the design of manufacturing cells is the identification of machine and component groups. This identification process requires an effective approach to form part families so that similarity within a part family can be maximized. According to Selim et al. (1998), clustering analysis is the most frequently used method for MC design. However, because the cellular formation problem (CFP) is a NP-complete problem, there is still the challenge of creating an efficient clustering method. This paper deals with the use of a discrete particle swarm optimization algorithm in clustering problems for cell formations. The remainder of the paper is organized as follows. In Section 2, we discuss related works concerning clustering, machine grouping and particle swarm optimization techniques. In Section 3, the statement of the problem is presented. The proposed algorithm is presented in Section 4. Experimental results are presented in Section 5. Finally, we give some conclusions and suggest some lines for future research.

2. Clustering problem and machine grouping

The attempts to solve clustering or machine grouping problems can be classified into two groups. The first group consists of algorithms that attempt to determine the optimal solution, and the second one, the metaheuristics or approximate methods. In the recent years, different metaheuristic methods have been used to solve the cell-formation problem (Boctor, 1991; Chen & Srivastava, 1994) and, more recently, Wu, Chang, and Chung (2008) used Simulated Annealing. Venugopal and Narendran (1992) presented the first attempt to solve a cell-formation problem with the help of an evolutionary computation algorithm. Gupta, Gupta, Kumar, and Sundaram (1996) employed the same genetic representation of solutions as Venugopal and Narendran (1992) but followed a different multi-objective optimization approach for the simultaneous minimization of the total number of intercell–intracell moves and within-cell load variation. Aljaber, Baek, and Chen (1997) and Lozano, Adenso, Salinas, and Giménez (1999) used Tabu Search.

Synergy effects between metaheuristics such as Simulated Annealing (SA) and GA have been commented on and demonstrated by Wu et al. (2008). James, Brown, and Keeling (2007) presented a hybrid grouping genetic algorithm for the cell-formation problem that combines a local search with a standard grouping genetic algorithm to form machine-part cells. Nsakanda, Diaby, and Price (2006) proposed a solution methodology based on a

* Corresponding author.
E-mail addresses: orlando.duran@ucv.cl (O. Durán), nibaldo.rodriguez@ucv.cl (N. Rodriguez), lac@upf.br (L.A. Consalter).

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.06.061

combination of a genetic algorithm and large-scale optimization techniques. Dimopoulos (2006) introduced multi-objective GP-SLCA, an evolutionary computation methodology, for the solution of the multi-objective cell-formation problem. GP-SLCA is a hybrid algorithm comprising GP-SLCA, a genetic programming algorithm for the solution of single-objective cell-formation problems, and NSGA-II, a standard evolutionary multi-objective optimization technique.

More recently, the paper of Andres and Lozano (2006) presented the first particle swarm optimization (PSO) algorithm designed to address the problem of Manufacturing Cell Formation. PSO is an evolutionary computation (EC) method inspired by flocking birds (Kennedy et al., 1995) and has been applied to many different areas, including manufacturing. PSO is initialized with a population of random solutions, where this initial population evolves over generations to find optimal solutions. However, in PSO, each particle in population has a velocity, which enables them to fly through the problem space instead of dying or mutating. Therefore, each particle is represented by a position and a velocity. The modification of the position of a particle is performed by using its previous position information and its current velocity. Each particle knows its best position (personal best) so far and the best position achieved in the group (group best) among all personal bests. These principles can be formulated as:

$$v_i^{n+1} = wv_i^n + c_1 r_1^n (p_i^n - x_i^n) + c_2 r_2^n \left(p_g^n - x_i^n\right), \tag{1}$$

$$x_i^{k+1} = x_i^k + v_i^{n+1}, \tag{2}$$

where w is inertia weight; c_1, c_2 are two positive constants, called cognitive and social parameters, respectively; $d = 1,2,\ldots,D$; $i = 1,2,\ldots,m$, and m is the size of the swarm; r_1, r_2 are random numbers, uniformly distributed in $[0,1]$; $n = 1,2,\ldots,N$, denotes the iteration number; and N is the maximum allowable iteration number.

The first term on the right hand side of Eq. (1) is the previous velocity of the particle, which enables it to fly in the search space. The second and third terms are used to change the velocity of the agent according to pbest and gbest. Generally speaking, the set of rules that govern PSO are: evaluate, compare, and imitate. The evaluation phase measures how well each particle (candidate solution) solves the problem at hand. The comparison phase identifies the best particles. The imitation phase produces new particles based on some of the best particles previously found. These three phases are repeated until a given stopping criterion is met. The objective is to find the particle that best solves the target problem.

3. Problem statement

In this work, clustering problem is considered as an optimization one. Thus, let us introduce the elements of this optimization model.

Given an incidence matrix $A = [a_{ij}]$, where:

$a_{ij} = 1$ if jth component visits ith machine;
$a_{ij} = 0$ otherwise.

When a machine-component incidence matrix is constructed, no machine groups or part families are easily visible. The main objective in machine grouping is the formation of set of machines and workpieces in groups so that the number of intercell transportations of pieces is minimized. Therefore, the initial matrix (Fig. 1a) has to be transformed into a matrix that has a block diagonal structure (Fig. 1b). This rearrangement aims at minimization of total intercell moves and of within-cell load variation. A rigorous mathematical formulation of machine-component grouping problem with these objectives is given by Boctor (1991).

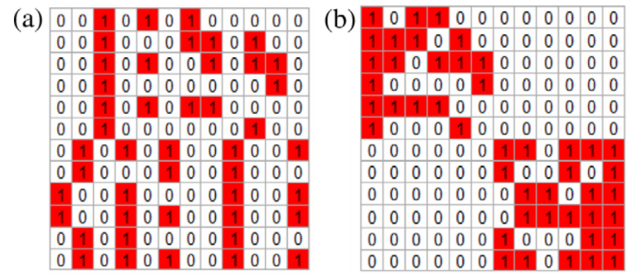

Fig. 1. Initial incidence matrix (a) and a rearranged matrix (b).

The optimization model is stated as follows. Let:

- M be the number of machines,
- P, the number of parts,
- C, the number of cells,
- i, the index of machines ($i = 1,\ldots,M$),
- j, the index of parts ($j = 1,\ldots,P$),
- k, the index of cells ($k = 1,\ldots,C$),
- $A = [a_{ij}]$, the $M \times P$ binary machine-part incidence matrix,
- $M_{\max}$, 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

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

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

The problem is represented by the following mathematical model:

$$\text{Minimize} \quad \sum_{k=1}^{C} \sum_{i=1}^{M} \sum_{j=1}^{P} a_{ij} z_{jk} (1 - y_{ik})$$

$$\text{Subject to} \quad \sum_{k=1}^{C} y_{ik} = 1 \quad \forall i,$$

$$\sum_{k=1}^{C} z_{jik} = 1 \quad \forall j,$$

$$\sum_{i=1}^{M} y_{ik} \leqslant M_{\max} \quad \forall k.$$

4. Proposed swarm-based clustering method

Traditional PSO algorithm was developed for continuous domains. A discrete binary version of the PSO algorithm was developed by Kennedy et al. (1995). Correa, Freitas, and Johnson (2006) proposed a Discrete PSO (DPSO) algorithm for attribute selection in data mining applications. The algorithm proposed here is based on the DPSO and the concept of proportional likelihoods, also used by Correa et al. (2006). The main difference between the traditional PSO algorithm and the algorithm proposed here is that the proposed algorithm does not use a vector of velocities as the standard PSO algorithm does. It works with a mechanism inspired by the proportional likelihoods concept. According to Correa et al. (2006), the notion of proportional likelihood used in the DPSO algorithm and the notion of velocity used in the standard PSO are somewhat similar.

This algorithm deals with discrete variables (cells), and its population of candidate solutions contains particles of a given size (n = number of machines). Each component of the particle (vector)