A Hybrid Algorithm Based on PSO and SA and Its Application for Two-Dimensional Non-guillotine Cutting Stock Problem*

J.Q. Jiang[1, 2], Y.C. Liang[1, 3**], X.H. Shi[1], and H.P. Lee[3]

[1] College of Computer Science and Technology, Jilin University, Changchun 130012, China
ycliang@public.cc.jl.cn
[2] College of Mathematics and Computer Science, National University of Inner Mongolia, Tongliao 028043, China
[3] Institute of High Performance Computing, Singapore 117528, Singapore

Abstract. In this paper we present a hybrid algorithm based on Particle Swarm Optimization (PSO) and Simulated Annealing (SA) approaches and apply it to two-dimensional non-guillotine cutting stock problem. The probability of trapping at the local optimum during the searching process can be reduced using the hybrid algorithm. Meanwhile, we propose a converting approach which is similar to the Bottom Left (BL) algorithm to map the cutting pattern to the actual layout. Finally, we implement the proposed algorithm on several test problems. The simulated results show that the performance of the hybrid algorithm is better than that of the standard PSO.

1 Introduction

The two-dimensional cutting stock problem can be stated as cutting small rectangular pieces of predetermined sizes from a large rectangular plate (the stock plate). Non-guillotine cut means that the cuts need not go from one edge of a rectangle to the opposite edge in a straight line. The aim is to minimize the unused area. Gilmore and Gomory [1, 2] used linear programming to solve such kind of a problem exactly. But because of the complexity of this problem, the exact algorithm only fits the case in which fewer pieces are being cut. These algorithms would fail if there are more pieces to be cut. A problem of cutting more than 20 pieces would cause some difficulty. Recently, with the extended application of various heuristic searching algorithms, Evolutionary Algorithms (EA) and Simulated Annealing (SA) approach have been applied to this kind of problems [3-5]. In this paper, we will consider applying Particle Swarm Optimization (PSO) to cutting stock problem. PSO was introduced by Kennedy and Eberhart [6]. In the process of searching for the solution, we add SA technique to PSO to reduce the probability of trapping at a local optimum.

---

* Supported by the science-technology development project of Jilin Province of China (Grant No. 20030520), the Key Science-Technology Project of the National Education Ministry of China (Grant No. 02090) and the doctoral funds of the National Education Ministry of China.

** To whom correspondence should be addressed. E-mail: ycliang@public.cc.jl.cn

M. Bubak et al. (Eds.): ICCS 2004, LNCS 3037, pp. 666–669, 2004.

2 A Hybrid Algorithm Based on PSO and SA

PSO is an evolutionary computational model based on swarm intelligence. Suppose that the search space is D-dimensional and m particles form the colony. The ith particle represents a D-dimensional vector X_i ($i=1, 2, \ldots, m$). It means that the ith particle locates at $X_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$ ($i=1, 2, \ldots, m$) in the search space. The position of each particle is a potential solution. We calculate the particle's fitness by putting its position into a designated objective function. When the fitness is higher, the corresponding X_i is "better". The ith particle's "flying" velocity is also a D-dimensional vector, denoted as $V_i = (v_{i1}, v_{i2}, \ldots, v_{iD})$ ($i=1, 2, \ldots, m$). Denote the best position of the ith particle as $P_i = (p_{i1}, p_{i2}, \ldots, p_{iD})$, and the best position of the colony as P_g $(p_{g1}, p_{g2}, \ldots, p_{gD})$, respectively. The PSO algorithm could be performed using the following equations

$$V_i(k+1) = w\, V_i(k) + c_1 r_1 (P_i - X_i(k))/\triangle t + c_2 r_2 (P_g - X_i(k))/\triangle t \qquad (1)$$

$$X_i(k+1) = X_i(k) + V_i(k+1)\triangle t \qquad (2)$$

where $i=1, 2, \ldots, m$, k represents the iterative number, w is the inertia weight, c_1 and c_2 are learning rates, r_1 and r_2 are random numbers between 0 and 1, Δt is the time step value, $V_i \in [V_{min}, V_{max}]$ where V_{min} and V_{max} are the designated vectors. The termination criterion for the iterations is determined according to whether the maximum generation or a designated value of the fitness of P_g is reached [7].

Simulated annealing is a local search algorithm. The searching process starts with an initial solution. A neighbor of this solution is then generated and the change of cost is calculated. For a general local search process, if a reduction of cost is found then the current solution is replaced by the generated neighbor. Otherwise, the current solution is retained. The process is repeated until no further improvement can be found in the neighborhood of the current solution [4].

In order to reduce the probability of trapping at a local optimum, we introduce the SA technique to the PSO. We use PSO to search for the initial best position firstly. If the best position P_g does not change for a specified generation, then the SA is used, which starts with P_g for a local search. This process is repeated until the termination condition is achieved.

3 Stock Cutting Algorithm

For the sake of simplicity, we assumed that all the pieces have fixed orientation and all cuts on the stock plate are infinitesimally thin. The pieces cutting from the stock plate are rectangular in shape. We describe the stock plate and the pieces in a free coordinates. The left bottom corner of the stock plate is placed at the origin. Each piece is denoted by a four-dimension vector (x_k, y_k, l_k, w_k), where (x_k, y_k) is the position of the left bottom corner of the piece on the stock plate, l_k and w_k are the length and width of the piece, respectively. Each particle in the colony denotes a cutting pattern. A cutting pattern is consisted of a set of pieces.

In PSO, the fitness function of a particle is taken as the ratio of the summed areas of the pieces completely placed on the stock plate to the total area of the stock plate.

In the SA the objective function of a particle is the trim loss which is the ratio of the lost area of the stock plate to the area of the stock plate.

The cutting algorithm deals with the particles (the cutting pattern). In order to place the pieces on the stock plate, we should convert the cutting pattern to an actual layout. In this paper, we present a converting approach which is similar to the Bottom Left (BL) algorithm [5]. We call it the Coordinate-based Bottom Left Algorithm (CBL). In the BL algorithm, the piece is firstly put at the right upper corner of the stock plate and then it is moved to the left. In this paper, in order to use the coordinates of the piece's left bottom corner, we place the piece according to the coordinate then move it. We sort the pieces on x_k to reduce the probability of overlapping.

Table 1. Simulated results for the five test cases

Number of pieces	Trim loss of PSO (%)	Trim loss of hybrid algorithm (%)	Size of stock plate
10	0	0	100×80
10	0	0	40×20
15	7.5	3.7	40×20
20	6.3	3.4	40×40
30	7.1	5.5	65×45

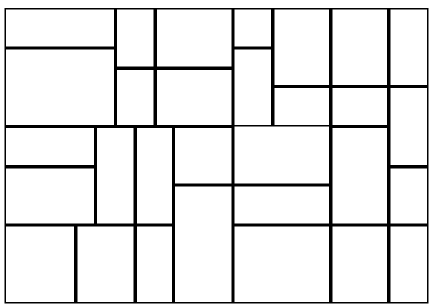

Fig. 1. A test problem with 30 pieces to be cut from the stock plate

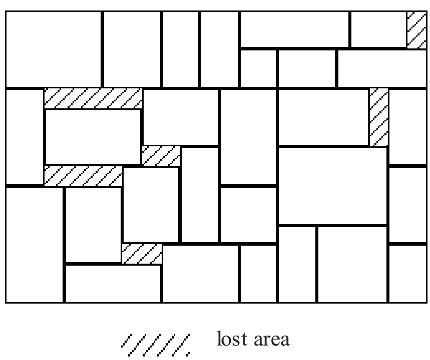

Fig. 2. Cutting results using the hybrid algorithm

4 Conclusions and Discussions

Five test problems are used to examine the proposed hybrid algorithm. Each of the five test problems has its own optimal solution of zero trim loss. Thus we can estimate easily the performance of the algorithm. The number of the rectangular pieces in each stock plate ranges from 10 to 30. The population size of the colony is taken as 60, and the parameters in the PSO are taken as $w = 0.3$, $c_1 = 0.8$ and $c_2 = 0.9$. The initial and final temperatures are taken as 80 and 1, respectively, in the SA. The algorithms are written in C and run in a Pentium 4 personal computer with 2.0GHz. The maximum number of iterations is taken as 1000.

Figure 1 shows one of the five test problems, where 30 pieces is cut from the stock plate. Figure 2 shows the actual layout generated using the hybrid algorithm. Twenty-eight pieces are cut from the stock pieces. The shade in the stock plate represents the lost area.

Table 1 shows the simulation results for the five test cases. From the results of the test cases it can be seen that the hybrid algorithm based on the PSO and the SA proposed in this paper has better ability to search for the global optimum for cutting stock problem. When the number of the pieces is smaller, both PSO and the hybrid algorithm work well. With lager number of pieces, the performance of the hybrid algorithm is better than the PSO's.

References

1. P. C. Gilmore and R. E. Gomory: A Linear Programming Approach to the Cutting Stock Problem. Operations Research, Vol.9 (1961) 849-859
2. P. C. Gilmore and R. E. Gomory: Multistage Cutting Stock Problems of Two and More Dimensions. Operations Research, Vol.13 (1965) 94-120
3. Stefan Jokobs: On Genetic Algorithms for the Packing of Polygons. European Journal of Operational Research, 88 (1996) 165-181
4. K. K. Lai and W. M. Chan: Developing a Simulated Annealing Algorithm for the Cutting Stock Problem. Computer and Industrial Engineering, Vol.33 (1997) 115-127
5. T. W. Leung, C. H. Yung and Marvin D. Troutt: Applications of Genetic Search and Simulated Annealing to the Two-dimensional Non-guillotine Cutting Stock Problem. Computer and Industrial Engineering, 40 (2001) 201-214
6. J. Kennedy, R. C. Eberhart: Particle Swarm Optimization. Proc. IEEE Conf. on Neural Networks, IV, Piscataway, NJ (1995) 1942–1948
7. X. H. Shi, L. M. Wan, H. P. Lee, X. W. Yang, L. M. Wang and Y. C. Liang: An Improved Genetic Algorithm with Variable Population-size and A PSO-GA Based Hybrid Evolutionary Algorithm. Second International Conference on Machine Learning and Cybernetics (2003) 1735-1740