Optimal Cutting of Two-dimensional Free Patterns Using Genetic Algorithms

Toshihiko Ono and Tomoyuki Yokoi
Dept. of Computer Science and Eng.,
Fukuoka Institute of Technology
3-30-1 Wajiro-higashi, Higashi-ku,Fukuoka 811-0295, Japan
e-mail: ono@cs.fit.ac.jp
http://www.fit.ac.jp/˜ ono/

Abstract

This paper describes a new method of genetic algorithms(GAs) to determine the optimal layout of a given number of free patterns on a sheet, to make the required length of the sheet minimal. At the arrangement, the patterns can be rotated in some cases and it is taken into considerations in the algorithms. By incorporating layout determing algorithms(LDAs) into GAs, this two dimensional search problem is handled as a one-dimensional problem and the calculation time required is reduced. The results of simulation studies meet our expectation.

1 Introduction

In this paper we are proposing a new method for the optimal cutting, in which a set of various two-dimensional patterns are cut from a sheet, to make the required sheet length minimal. Concerning the two-dimensional optimal cutting, there are two kinds of problems, the optimal cut of rectangular patterns and that of free patterns. The optimal cutting of rectangular pieces from a sheet has been studied by many researchers[1, 2, 4, 13]. On the other hand, as to the optimal cutting of free patterns, which will be required in the cutting processes of clothes, metal sheets and the like, there have been not so many studies[7, 12], because of the difficulty of theoretical treatment. After much consideration, we have succeeded to solve it by GAs which incorporate special algorithms.

The given problem is originally a kind of two-dimensional search. As the space of two-dimensions is defined by the product of each dimension, it becomes quite large compared with that of one-dimension. Therefore, with the application of GAs to this problem, it is difficult to get proper results with a usually adopted tactic, in which genes represent the two-dimensional positions of patterns and the search is done two-dimensionally. In the method adopted here [11] the problem is solved as a one-dimensional ordering by incorporating LDAs into GAs to reduce the dimension of search space. At the same time, LDAs are designed to get the solution as quickly as possible.

In addition, considering when the sheet is homogeneous, such as plastics and steel etc., the rotation of each part is taken into consideration at the arrangement, to save the required sheet more.

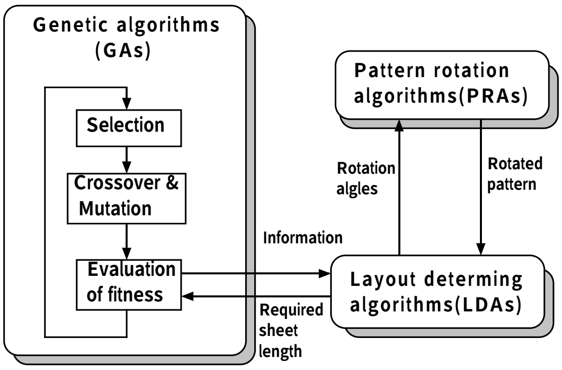

Figure 1: Configuration of systems

We have conducted simulation tests to confirm the performance of this method.

2 Principle

In this section, the principle of the method is explained. A set of various two-dimensional free patterns are to be arranged on a sheet to satisfy the following two requirements, when these patterns are produced by cutting the sheet.

1. To lay them out on the sheet without mutual overlapping,
2. To make the required sheet length minimal.

In applying GAs to this kind of problem, the normally adopted representation of genes will be the position of each pattern on a sheet. From the estimation of the size of search space, however, we have concluded that this representation method is impractical. In this method, since the genes will be a binary representation consisting of both the positions in width and those in length to all of the patterns, the length of the genes is $N(\log_2 W + \log_2 L) = N \log_2(WL)$ and thus the size of search space becomes $2^{N \log_2(WL)} = W^N L^N$, where N is the number of patterns, and W and L are the width and length of the sheet respectively.

On the other hand, if it is possible to solve the problem as an ordering one, the size of search space becomes $N!$, that is, the permutation of N. Considering the amount of W,L,N, we can easily understand that a considerable reduction of search space can be obtained with this method. From these results, we have decided to solve the problem by an ordering GAs, introducing LDAs to change a two-dimensional search into a one-dimensional ordering one.

Figure 1 shows the flow chart of the systems. The GAs and LDAs work sequentially, communicating each other. When the sheet is non-directional, it is possible to rotate patterns at arrangement. Considering these cases, we have included pattern rotation algorithms(PRAs) to this system. The GAs determine the order of patterns to arrange them on a sheet and send it to the LDAs. The LDAs fix the position of each pattern according to the order given by the GAs so that each pattern on the sheet doesn't have an overlap and an extra opening, after it is rotated by the angle given by the genes at the PRAs. When the layout of all the

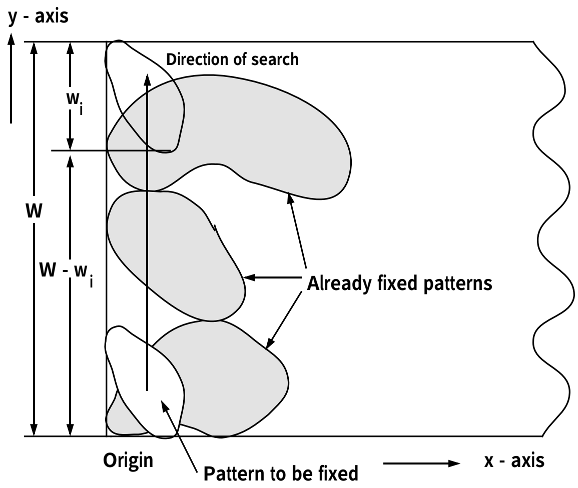

Figure 2: Search of overlapping

patterns has been fixed, the LDAs return the required sheet length to arrange the patterns to the GAs. With this length data, the GAs determine the order of pattern arrangement to make the object value(the required sheet length) minimal, by genetic operations such as selection, crossover and mutation.

3 Layout algorithms

To determine the position of each pattern on a sheet without overlapping and extra openings between patterns according to the order given by the GAs, we have introduced layout determining algorithms. As shown in Figure 2, the allowable position for each pattern is searched on the sheet, each starting from the origin of the sheet and walking in the direction of width(Y-axis), while stepping in the longitudinal direction(X-axis). The allowable position being found, the pattern is set there and a marking is given to the sheet data to store the occupied area by the pattern. To minimize the required search time, the followings are considered.

Let pattern-A be the pattern to be searched for its position on the sheet. At first the check is done to see if there is an overlap between the pattern-A and those already fixed on the sheet. Even though this check is originally two-dimensional, it can be done under an one-dimensional basis or less, by checking overlapping on the boundary, as shown in Figure 3. There is no overlap, as long as no part of the boundary is inside of already fixed patterns on the sheet. Therefore, by checking if the boundary is inside the already fixed patterns, the necessary overlap-check can be made and time saving is attained because this checking is done along the boundary of one-dimension. Moreover, if at least one point on the boundary is within the fixed patterns, it means that there is an overlap. By selecting proper points on the boundary fewer than the points of the boundary and checking at these points prior to the boundary contributes to another savings in calculation time. As Figure 4

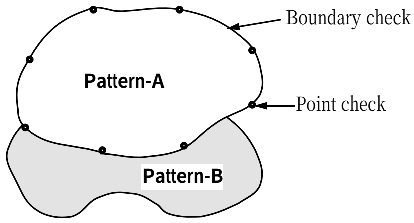

Figure 3: Check of overlapping

shows, the check at these points and the same at the boundary are integrated to act serially in the LDAs. If one of them fails, the pattern-A jumps to the next search position by an amount equal to the distance calculated by the following method. After both of these checks have succeeded, the pattern-A is set there.

When an overlap is found, the distance necessary to jump to the position where no overlap exists is calculated, referring to the mutual position between the pattern-A and each of the overlapped fixed patterns on the sheet. As shown in Figure 5, the method is divided into two methods according to the mutual position. At first, when the point P on the upper boundary of the pattern-A is in the the pattern-B as shown in (a)of Figure 5, the jump distance Δy of the pattern-A to get out of the pattern-B in the direction of the Y-axis are sum of Δy_1 and Δy_2, because the portion of the pattern-A between the points P and Q should be out of the pattern-B. The distance Δy_1 is obtained by scanning upward from the point P to the boundary of the pattern-B. On the other hand, since Δy_2 is inherent to the point P, it can be calculated beforehand and stored in memory. As for the points on the lower boundary, Δy_2 is zero as can be understood from (b) of Figure 5, hence only the calculation of Δy_1 is required. The required jump distance of the pattern A is calculated as the maximum value of Δy for all the points on the boundary of the pattern-A.

The flow chart also shows the procedures at the near end of the sheet in the width side. In the scanning in the Y-direction, it is unnecessary to scan up to the upper end of the sheet, but it is enough to scan for the range reduced by as much as the width of the pattern-A.

4 Genetic algorithms

As to the application of GAs to an ordering problem, various methods have been studied. Among them, we have used Genitor [3][8] as a software workbench of GAs. The genes are constructed as a dual chromosome construction, that is, two groups of genes. One group of genes are used to define the order of the part arrangement and expressed by order numbers, and another group is for the rotation of parts. Different genetic operations are applied for each group. For the ordering part, the genes are represented by the path representation, with the rank method for the selection. For the crossover, three methods have been compared among various methods applied to the path representation: CX(Cyclic crossover), PMX(Partially-mapped crossover) and OX(Order crossover)[8]. As to the mutation, the method exchanging one random element of genes with another random one is added to Genitor.

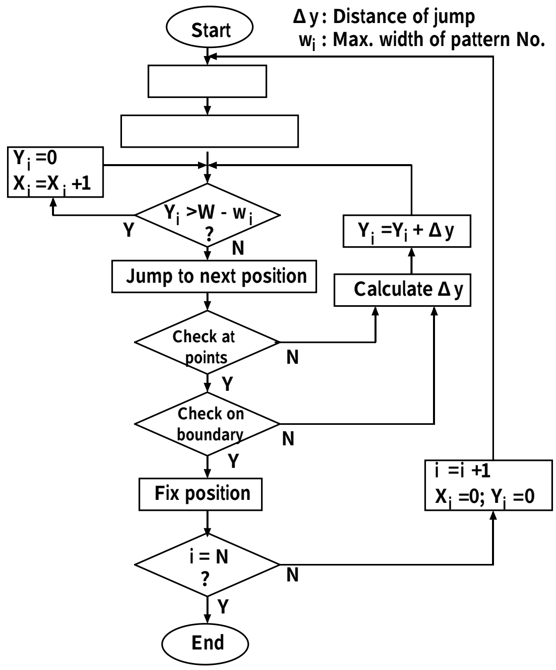

Figure 4: Flow chart of LDAs

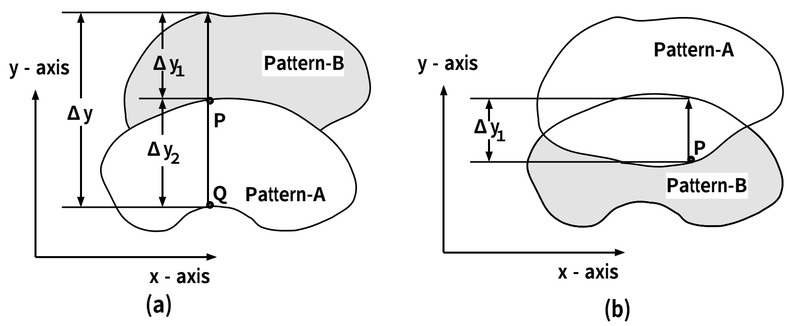

Figure 5: Calculation of jump distance

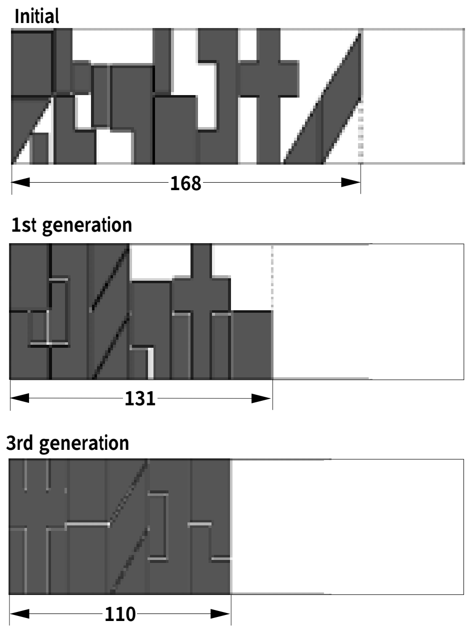

Figure 6: Optimal arrangement of 14 patterns

For the rotation part, the genes consist of integers expressing the degree of rotation and their crossover and mutation are performed at the same time when the crossover of the order part is done.

5 Results of simulation

We have conducted various simulations to confirm the performance of this method and the followings are some of them. The first simulation is to test whether this method can get the optimal solution, taking the case where the optimal value is obvious as shown in Figure 6. The optimal arrangement of a set of 14 patterns has been obtained at the third generation, with 50 individuals and starting from a random state.

The second is for the optimal layout of free patterns as shown in Figure 7, to investigate the performace for the free patterns and the effect of the rotation of parts on the layout. The figure (a) shows the arrangement gained at 400th generation with the rotation of parts, and figure (b) without the rotation. Figure 8 shows the comparison of convergence characteristics for both cases. As there is no theoretical method to determine that the results gained here are optimal, we can only estimate it by simulations. Referring to various results gained, at least the near optimal layout seems to be gained.

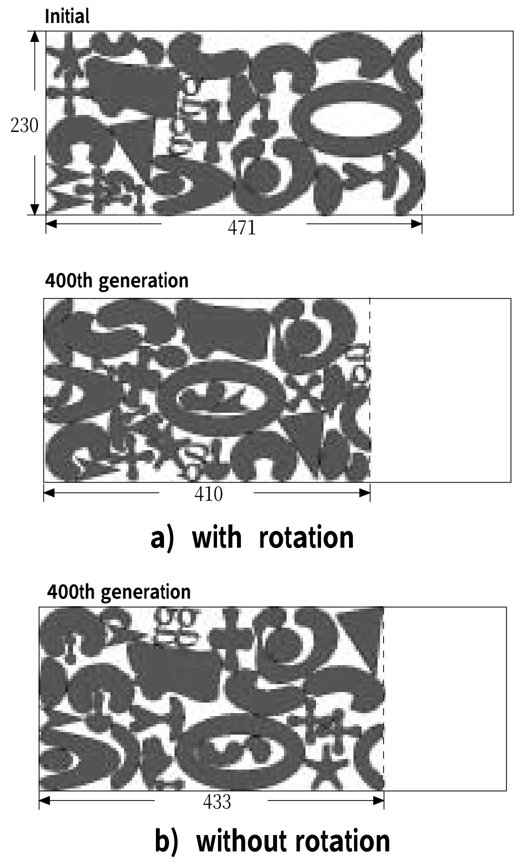

Figure 7: Optimal arrangement of free patterns

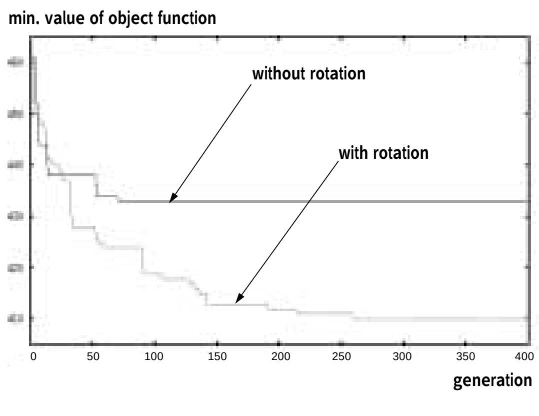

Figure 8: Comparison of convergence characteristics

6 Conclusions

In this paper we have explained an application of GAs to the optimal cutting of two dimension free patterns from a sheet. By combining GAs with layout determining algorithms a two-dimensional problem is solved as an one-dimensional order one in a short time.

References

[1] Chauny, F. and Loulou, R. et al. : A Two-phase Heuristic for the Two-dimensional Cutting-stock Problem, Journal of the Operational Research Society, Vol.42, No.1, pp.39–47, 1991

[2] Christofides, N. and Hadjiconstantinou, E. : An exact algorithm for orthogonal 2-D cutting problems using guillotine cuts, European Journal of Operational Research 83, pp.21–38, 1995

[3] Davis, L. Ed. : Handbook of Genetic Algorithms, Van Nostrand Reinhold, 385p, 1991

[4] Dyson, R. G. and Gregory, A. S. : The Cutting Stock Problem in the Flat Glass Industry, Operational Research Quarterly, Vol.25, No.1, pp.41–53, 1974

[5] Goldberg, D.E. : Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Pub. Co., 412p, 1989

[6] Holland, J.H. : Adaptation in Natural and Artificial Systems, The University of Michigan Press, 211p, 1975

[7] Jakobs, S. : On genetic algorithms for the packing of polygons, Proc. of the KI-94 Workshop, pp.84–85, 1994

[8] Michalewicz, Z. : Genetic Algorithms + Data Structures = Evolution Programs, Second, Extended Edition, Springer-Verlag, 340p, 1994

[9] Ono, T. and Watanabe, G. : Application of Genetic Algorithms to Optimizing Problems(In Japanese), Language and Information Processing (Fukuoka Institute of Technology), Vol.6, pp.89-96, 1995

[10] Ono, T. and Watanabe, G. : Application of Genetic Algorithms to Optimal Selection of Cutting Length of Bars, Proceedings of Fifth FIT-Ajou University Joint Seminar, pp.24-31, 1995

[11] Ono, T. and Watanabe, G. : Gnetic Algorithms for Optimal Cutting, in Evolutionary Algorithms in Engineering Applications, ed. by D.Dasgupta and Z.Michalewicz, Springer-Verlag, pp.515-530. 1997

[12] Petridis, V. and Kazarlis, S. : Varying Quality Function in Genetic Algorithms and The Cutting Problem, Proc. of the First IEEE Conf. on Evolutionary Computation, pp.166–169, 1994

[13] Vasko, F. J. : A computational improvement to Wang's two-dimensional cutting stock algorithm, Computers ind. Engng, Vol.16, No.1, pp.109–115, 1989

[14] Whitley, L.D. and Vose, M.D. Ed. : Foundations of Genetic Algorithms.3, Morgan Kaufmann Pub., 336p, 1995