

---

# Nesting of Irregular Shapes Using Feature Matching and Parallel Genetic Algorithms

---

Anand Uday Erik D. Goodman      Ana      nda A. Debnath  
Genetic Algorithms Research and Applications Group (GARA      Ge)  
Michigan State University  
2857 W. Jolly Road, Okemos, MI 48864  
goodman@egr.msu.edu

## Abstract

The problem of finding a dense packing of a set of two-dimensional polygonal shapes within another larger two-dimensional polygon is called nesting. This problem finds wide application in the manufacturing, leather cutting, and textile industries—in short, where material is costly so scrap must be minimized. In this paper, we describe a new approach to solving this problem. It is a hybrid (or memetic) approach, which uses a parallel genetic algorithm and a heuristic based on shape information in the form of feature matching. In our experiments with the parallel GA, we tried various topologies for the communication among subpopulations, and various migration policies. A good choice of communication patterns seems to give subpopulations more time to explore by themselves before they are “contaminated” by individuals from other subpopulations, while still allowing for useful sharing of building blocks gained. Our test problems show this approach to work well in this type of problem, where the search domain is very large.

## 1 INTRODUCTION

Layout and cutting problems are important in many industries, as they involve the optimal use of valuable material. Problems of optimal arrangement of 2-D pieces to be cut from an initial piece of stock material are called nesting problems, and there are many varieties, depending on the shapes of the pieces, constraints on their orientations, etc. The problem to be addressed in this paper can be stated as follows: given a rectangular piece of stock of a specified width and indefinite length, find the optimal arrangement of a given set of polygonal “part” shapes so that a) none of the parts overlaps any others, b) all are contained within the boundary of the stock piece, and c) the length of the stock piece used is minimized. In this case, there is no constraint on the orientation of the part shapes, but they may not be “turned over.” Note that there is no

constraint, such as convexity, on the shape of the polygonal parts.

In recent years, a number of researchers have investigated the problem of nesting of irregular shapes. The heuristic approaches taken to solve this problem can be broadly classified into two categories: rule-based approaches and stochastic algorithms. In a rule-based heuristic, a set of rules is designed to try to take advantage of some characteristics of the shapes of the parts, placing earliest those with certain characteristics, packing together parts with certain matching features, etc. On the other hand, stochastic approaches such as genetic algorithms or simulated annealing, typically use little information about part shapes, instead using only simple “packing” rules and relying on the stochastic algorithm to vary the order in which these rules are applied to the parts to be nested.

The approach we have chosen is a hybrid one with strong reliance on a powerful feature-matching heuristic capable of generating fairly good nestings even without a genetic algorithm. It uses part shape features to determine the exact placement and orientation of the parts, here augmented by a genetic algorithm that determines the sequence in which they are nested (now sometimes together called a “memetic” algorithm). We have used a parallel GA to make the search both more global and more efficient.

## 2 RELATED WORK

A significant amount of research has been done in this area. Freeman and Shapira (1975), and Adamowicz and Albano (1976) have approached this problem by shape approximation, in which they approximate the shape by their minimum-enclosing rectangle and then pack the rectangles. To overcome the waste associated with the rectangular approximation, attempts have been made to first generate pairwise rectangular clusters, if the shapes allowed (Nee *et al.*, 1986).

Several attempts have also been made to solve this problem using local greedy search heuristics. Albano and Sapuppo (1980) proposed an algorithm that decided the next part to be placed by evaluating the potential waste due to the placement of the part at hand.

In 1996, Lamousin *et al.* proposed an algorithm that modifies Albano and Sapuppo’s algorithm, using the

concept of No Fit Polygon (NFP) for part placement. Lamousin and Waggenspack (1996) also proposed another algorithm that uses features of the profile. This algorithm matches complementary features of the part and the remaining area of the stock.

Besides the deterministic approach, probabilistic and evolutionary techniques have been used to solve this problem. G-C Han and S-J Na (1996) used a two-stage method with a neural-network-based heuristic for generating an acceptable initial layout, and a simulated annealing algorithm for fine-tuning the solution. In 1995, Ismail and Hon proposed a genetic-algorithm-based solution to this problem. They generated a set of initial random layouts as their first generation chromosomes. These layouts were allowed to have parts overlapping each other. They constructed an objective function that tried to minimize the total area needed to nest them and included a penalty associated with the overlaps. Jain and Gea (1998) proposed a solution based on genetic algorithms by using a 2-D representation for the chromosomes. Babu and Babu (1999) described a genetic algorithm which aimed at finding an optimum sequence in which the parts are to be placed on the sheet (Babu and Babu, 1999). In 2000, Sha and Kumar came up with a representation that encoded the sequence and the orientation of the part on a 2-D chromosome and modified the genetic algorithm operators to deal with that form.

We describe here an approach that uses a genetic algorithm wedded with a powerful heuristic to solve the problem more effectively.

### 3 PARALLEL GENETICAL ALGORITHMS

Genetic algorithms are often useful in solving highly multimodal problems, of which nesting is a fine example. Although GA's can be made resistant to premature convergence, they are not immune. One technique to reduce the likelihood of premature convergence is parallelization of the GA using multiple subpopulations. The two most commonly used kinds of parallel GAs are the fine-grain GA and coarse-grain GA. In a fine-grain GA, small subpopulations (or individuals) are typically arranged over a grid-like space, and each interacts frequently (for breeding, etc.) with its immediate neighbors. In a typical coarse-grain (or "island") GA, each subpopulation occasionally sends migrants to a specified set of neighbors, but the communications topology is typically open and frequency of communication low.

In this paper, we have investigated the nesting problem using a coarse-grain parallel GA.

**Population Representation:** In a "classical" GA, a binary string representation was often used. However, for sequencing and other combinatorial problems, chromosomes often represent an ordering of entities, specified simply as a permutation of the integers  $\{0, 1, \dots, N-1\}$ . For this work, the chromosome is such a permutation, ranging over the indices of the parts to be

nested, for example: (3, 4, 5, 9, 7, 6, 8, 0, 2, 1). In this example for nesting of ten parts, the chromosome would be interpreted as the sequence in which the parts are to be considered and placed on the sheet by a heuristic nesting rule.

**Crossover:** For this representation we considered four different crossover operators: partially matched crossover (PMX), uniform order-based crossover (UOBX), order-based crossover (OBX) and cycle crossover (CX) (Davis, 1991; Goldberg, 1988).

**Mutation:** The mutation operators we tested here were swap mutation and scramble mutation (Davis, 1991; Goldberg, 1988).

**Crowding and Incest Reduction:** To reduce the chances of premature convergence, we made the use of a DeJong crowding factor (Goldberg, 1988). It helps in allowing several distinct groups of individuals to develop and persist in the population. This technique is useful when exploring problems that are strongly multi-modal. In addition, the mechanism of incest reduction (Goodman, 1994) reduces the proportion of crossovers performed between very similar chromosomes. This technique helps to maintain genetic diversity and thus helps in avoiding premature convergence.

**Elitism:** We used elitism to insure that at least one copy of the current generation's best individual appears in the next generation.

**Fitness Function:** The problem aims at minimizing the length used of a fixed-width piece of rectangular stock. However, in preliminary studies, we observed that nesting larger parts first often yielded a better solution. In order to speed the search and exploit this, in some of our runs, we added a bias term to our fitness function that slightly favored nestings in which larger (area) parts were placed first. However, the effect of this term was not found to be large, and we abandoned its use in the later runs. Linear scaling of fitnesses was used to control selection pressure (Goldberg, 1988).

## 4 SHAPE INFORMATION AND FEATURE MATCHING

In our hybrid (or memetic) algorithm, shape information is used to effectively match complementary features on the parts and the stock. In this case, building on the earlier work of one of the authors (Debnath, 1997), we have defined a feature to be an instance of two adjacent edges on a polygon. The data defining this type of feature are the lengths of the two adjacent edges and the internal angle between these edges. Figure 1 shows an example of such a feature.

### 4.1 Placement Policy

Given the next part to be nested, the algorithm determines what position and orientation is best for the part vis-à-vis the current state of the stock. At any point, the system tracks a "stock profile"—a polyline comprised of portions of edges of stock and parts, and that includes all currently

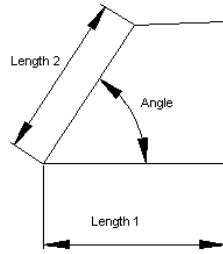


Figure 1: Feature information

open area in the stock as nested to date. Candidates for features at which to nest subsequent parts are located on this profile. The profile will grow to include many small, closed areas in which no additional parts can be nested, and the algorithm, after numerous attempts to nest subsequent parts in such an area, will eventually mark the points in that area as “bad” points, and will refrain from trying to nest more parts there.

The packing heuristic first selects the vertex on the profile (ignoring “bad” interior points) with the lowest co-ordinate. If more than one vertex has the same y-coordinate, the vertex with the smallest x-coordinate is selected. This selection is based on the placement policy of *lowest and if necessary leftmost*. The heuristic forms a target feature on the stock, and iterates through all the features of the part at hand. To each of the iterations through the part features it assigns a particular score. The orientation yielding the highest score is retained for the final placement. This score is calculated using the following parameters:

**Left Shadow Area** (see Figure 2): Since our placement policy was selected to fill up the stock from left to right, any closed-off areas to the left of the stock would be unfavorable for the final solution.

**Bottom Shadow Area** (see Figure 2): Since our placement policy was also set to fill up the stock from bottom to top, any closed-off area towards the bottom of the part would again be unfavorable for the solution.

**Contact Length of the Feature** (see Figure 2): To effectively exploit a corner feature, it was necessary to calculate the contribution due to the feature itself. To ensure good local packing, we wanted the contact length between the part at hand and existing stock profile to be maximized. Further, in order to yield comparable units in the scoring function, the value of this measure is squared.

The scoring function used is thus:

$$a * (\text{Left Shadow Area}) + b * (\text{Bottom Shadow Area}) + c * (\text{Contact Length})^2,$$

where  $a, b < 0$  and  $c > 0$ .

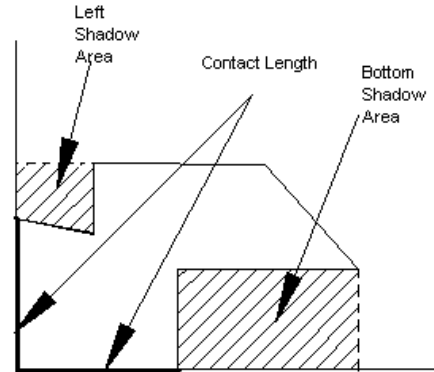


Figure 2: Sample part and stock illustrating some packing measures

## 5 TEST PROBLEMS AND RESULTS

In order to test our algorithm on realistic industrial examples, we took as the base problem a set of parts to be nested on steel plate stock in the shipbuilding industry. Complex geometry and varying sizes characterize the set of parts, possibly including multiple copies of the same part. We made several test runs to determine suitable operators, migration strategy, and values for various coefficients. Table 1 lists the experiments and findings for the genetic operators and best coefficients for score calculation.

Of all the crossover operators, uniform order-based crossover (UOBX) (see explanation below) worked best, both in sets of runs using each crossover operator for single runs and using different crossover operators for different subpopulations in the same run. Considering the nature of this problem and the manner in which UOBX works, this was expected. In the case of UOBX, there seemed to be a higher probability that meaningful building blocks were preserved.

Table 1: Experiments and findings

Experiments	Best Performance using:
Crossover Operators: UOBX, PMX, CX, OBX	UOBX
Mutation Operator: Swap, Scramble	Swap
Values of coeff. 'sa, b, c	a=-1, b=-1, c=2.5~3.0

Briefly, by example, the UOBX operator works as follows:

Step 1: Selection of parents

Parent 1: 123456789

Parent 2: 245796318

Step2:Randomgenerationofbinarytemplate

011001011

Step 3: The 1's specify loci to be filled by the corresponding alleles of the first parent in the first child and the 0's are filled on the second child with corresponding alleles of the second parent.

Child1(partial):-23--6-89

Child2(partial):2--79-3--

Step4: The void '- 'space of the first child is filled by genes of parent 2 in their order of appearance (without duplication, since the result must be a permutation) and the second child is handled correspondingly.

Child1:423576189

Child2:214795368

Swap mutation (exchanging alleles at two randomly chosen loci) was observed to yield better results than scramble, as was again expected for this problem.

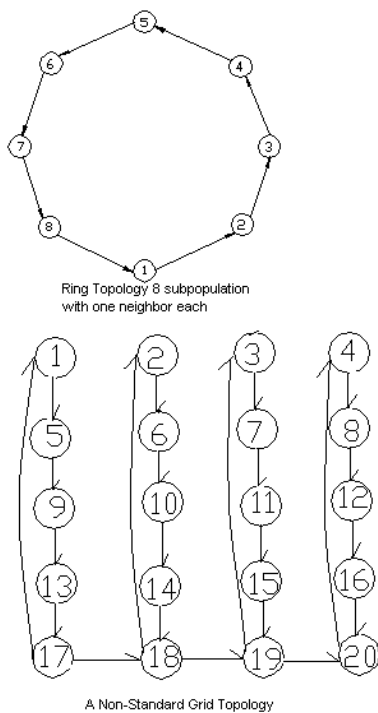


Figure3: RingandGridTopology

The set of values of coefficients given in Table 1 gives better performance than other combinations, although this search was not definitive. We arrived at this conclusion by assigning different sets of coefficients to different subpopulations in the same run. We observed that the subpopulation with  $a = -1$ ,  $b = -1$ , and  $c = 2.5 \sim 3.0$  generally yielded a better result. A possible implication of this trend is that matching of the contact lengths between two polygons is important and can result in better packing.

ld,

the

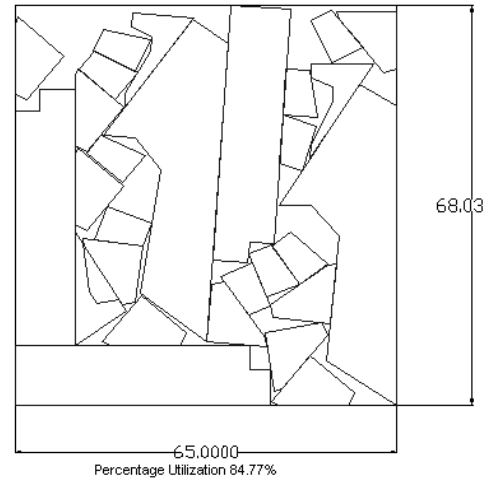


Figure4:Asampleoutputwith28parts

Preliminary runs were made with three sets of subpopulation sizes: 20, 50, and 200 per subpopulation, and 50 seemed to be sufficient. The crowding factor was set at 1/10 of the subpopulation size in all runs, as was the incest-reduction factor.

For this problem, we also tried different migration rules using different topologies. The topologies used are shown in Figure 3. Two types of topologies were tried: i) ring topology, ii) a variety of grid topologies. For the ring topology, we used eight subpopulations of 50 individuals each, with the single best individual of each subpopulation migrating to the nearest subpopulation in a counter-clockwise direction every ten generations. In the grid topology, the migration scheme was different. In this case, in order to try to provide a sufficient number of migrants to allow for significant influence on the recipient subpopulation, but to avoid essentially making the subpopulations a well-mixed single subpopulation, one tenth of each subpopulation was migrated to each of its neighboring subpopulations, but only every tenth generation. The neighboring subpopulations, each containing 50 individuals, were defined as shown in Figure 3. The design of this grid was an attempt to provide a diversity of communication path lengths among various subpopulations in the total set – i.e., within a vertical subring, subpopulations communicated relatively quickly, but between columns, communication was delayed. It was observed that this grid topology yielded better and more consistent results than the simple ring topology and high migration frequency. It may be due to that fact that the grid topology helped in maintaining the genetic diversity among subpopulations for more generations and thus gave the subpopulations more time to evolve. The topologies described above were implemented in GALOPPS (Goodman, 1996), a GNU-licensed freeware developed at MSU's GARAGE. Figures 4 and 6 show sample outputs of problems we tested. Figure 4 nested 28 parts, while for Figure 6, these parts were duplicated.

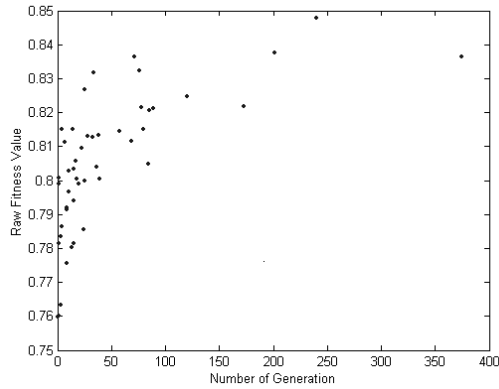


Figure 5: Plot showing each time a new best-of-population nesting is found in any subpopulation (exclusive of migration), versus generation when it occurred, for a sampler run using the grid topology of Figure 4.

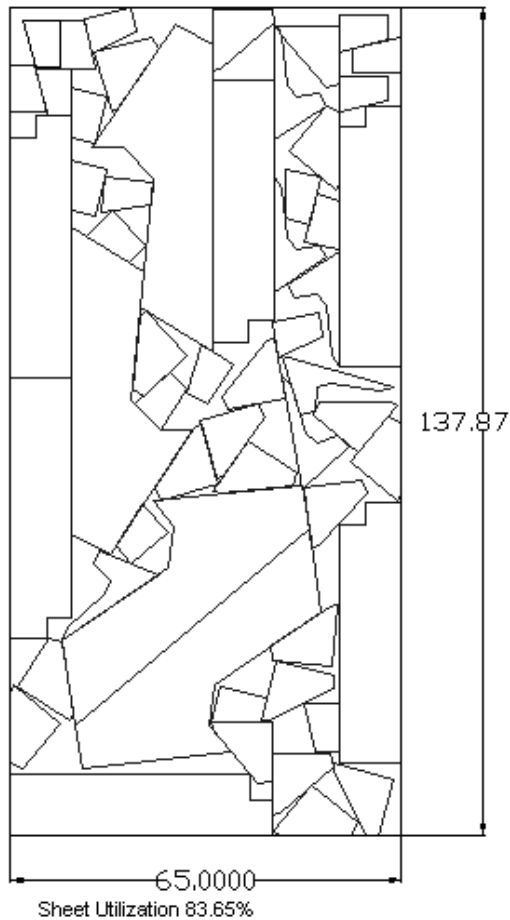


Figure 6: A sample output with 56 parts.

cated to yield a more challenging problem and to test the scalability of the algorithm. Looking at the percentage utilization of the sheet, we observe that the algorithm gave fairly consistent results, with run time scaling more

than linearly but less than quadratically with the number of objects to be nested. The GA usually stopped making rapid progress after about 200 generations. Figure 5 shows a plot of raw fitness value versus number of generations. The percentage utilization of stock typically obtained with this particular set of shapes was 83% - 84%.

## 6 SUMMARY AND CONCLUSIONS

This paper describes an application of a shape-feature-matching heuristic and a coarse-grain parallel GA to a challenging form of nesting problem. The use of shape information and feature matching helped in finding feasible solutions very effectively. It made the search more efficient by doing local search within each evaluation of the GA. We also tested an unusual grid topology and migration scheme. The results suggest that this led to improvement in performance of the GA. In industries such as shipbuilding, where the material is quite costly, even a half-percent improvement in packing density is sufficient to justify a fairly intensive search process, such as represented by the method described here. Of course, the computational intensity of this approach makes it inappropriate for "real-time" decision making on less costly nesting problems.

## References

- M. Adamovicz, and A. Albano (1976). "Nesting Two-Dimensional Shapes in Rectangular Modules," *Computer-Aided Design*, **8**(1):27-33.
- A. Albano, and G. Sapuppo (1980). "Optimal Location of Two-Dimensional Irregular Shapes Using Heuristic Search Methods," *IEEE Transactions on Systems, Man, and Cybernetics*, **10**(5):242-248.
- A. R. Babu and N. R. Babu (1999). "Effective Nesting of Rectangular Parts in Multiple Rectangular Sheets Using Genetic and Heuristic Algorithms," *International Journal of Production Research*, **37**(7):1625-1643.
- L. Davis (1991). *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York.
- Ananda A. Debnath (1997). *A New Approach to Solving 2-Dimensional Part Nesting and Layout Problems*, M.S. Thesis, Dept. of Mechanical Engineering, Michigan State University.
- H. Freeman and R. Shapira (1975). "Determining the Minimum Area Encasing Rectangle for an Arbitrary Closed Curve," *Communications of the ACM* **81**(7):409-413.
- D. E. Goldberg (1988). *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Reading, MA.
- Erik D. Goodman (1994). *An Introduction to GALOPPS, Release 2.35*, Technical Report GARAGe94-5, Genetic

Algorithms Research and Applications Group (GARAGe),MichiganStateUniversity.

Erik D. Goodman (1996) *An Introduction to GALOPPS, Release 3.2*, Technical Report GARAGe96-07-01, Genetic Algorithms Research and Applications Group (GARAGe),MichiganStateUniversity.

G. C. Han and S. J. Na (1996). "Two-Stage Approach for Nesting in Two-Dimensional Cutting Problems Using Neural Network and Simulated Annealing," *Proceedings of Institution of Mechanical Engineers, Part: B Journal of Engineering Manufacture* , **210**:509-519.

H. S. Ismail and K. K. B. Hon (1995). "The Nesting of Two-Dimensional Shapes Using Genetic Algorithms," *Proceedings of Institution of Mechanical Engineers, Part: B Journal of Engineering Manufacture* , **209**: 115-124.

S. Jain and C. H. Gea (1998). "Two-Dimensional Packing Problem Using Genetic Algorithm," *Engineering with Computers*, **14**:177-190.

H. J. Lamousin, W. N. Waggenspack, and G. T. Dobson (1996). "Nesting of Complex 2-D Parts within Irregular Boundaries," *Transactions of ASME: Journal of Manufacturing Science and Engineering* , **118**:615-622.

H. J. Lamousin and W. N. Waggenspack (1996). "Nesting of Two-Dimensional Irregular Parts Using a Shape Reasoning Heuristic," *Computer-Aided Design* , **25**: 221-237.

A. Y. C. Nee, K. W. Seow, and V. C. Long (1986). "Designing Algorithm for Nesting Irregular Shapes With and Without Boundary Constraints," *Annals of CIRP*, **35** (1):107-110.

O. P. Sha and R. Kumar (2000). "Nesting of Two Dimensional Irregular Parts Within an Irregular Boundary Using Genetic Algorithms," *Journal of Ship Production*, **1680**:232-242.