ELSEVIER

PII: S0010-4485(96)00065-6

Computer-Aided Design, Vol. 29, No. 3, pp. 221-238, 1997

0010-4485/97/$17.00+0.00

Practice & experience

Nesting of two-dimensional irregular parts using a shape reasoning heuristic

Henry Lamousin and Warren N Waggenspack, Jr

A technique is outlined for the allocation or 'nesting' of irregular parts into arbitrarily shaped resources. Placements are generated by matching complementary shapes between the unplaced parts and the remaining areas of the stock material. The part and resource profiles are characterized by varying levels of detail using geometric 'features' at each stage of processing to intelligently select and place parts of the resource. Practical implementation issues are described and the performance of the algorithm contrasted to previously published works. Data from a marine fabrication facility is used for testing. © 1997 Elsevier Science Ltd. All rights reserved.

Keywords: nesting, shape reasoning, features

INTRODUCTION

Cutting and packing problems encompass a broad spectrum of research with applications in many industries. These include the fabrication and preparation of goods such as textiles, leather, paper, lumber, glass, and sheet metal components to name a few. Many solution methods exist to deal with the associated characteristics of each problem. In *Cutting and Packing in Production and Distribution*[8], Dyckhoff proposes a typology to represent concisely the broad range of problems and solutions. This includes the relevant properties of the problem addressed in this paper which he classifies as a specific case of the Bin Packing Type. Using data from ship and offshore oil platform fabrication, the allocation or 'nesting' of two-dimensional irregular (non-rectangular) profiles is investigated. The assortment of profiles is variable and there is no orientation restriction on the placement of parts. Seldom do multiple identical copies of a part occur in a single job thus eliminating the benefits of part grouping strategies. The problem is further generalized by the requirement that, where possible, items be placed within irregular holes or void regions existing in some larger parts. All other resources or stock materials specified by the user are rectangular. Various sizes are specified but optimum stock selection is not addressed.

Algorithms for solving this class of problem may be placed into three broad groups. The first is rectangular approximation, where either individual or clusters of parts are represented using a rectangular enclosure. Once all pieces are resolved in this way, solutions are generated using the numerous techniques available for nesting rectangular forms. Probabilistic optimization techniques are the basis of the second category of solutions. Here, the minimum for a trim waste function based on part overlap and overall layout dimensions is found using multistart or simulated annealing methods. The third group, discussed below, contains the remaining class of solutions, including the research presented. These methods are alternately referred to as rule based, expert, or intelligent systems. Each group is summarized and additional information provided in References 7 and 13.

Rule based systems resolve the allocation problem using heuristics which attempt to mimic the approach of a human or manual nester. Reasoning is used to select and locate the next part to be placed at each stage of the solution. These methods are attractive when exhaustive examination of all available combinations is not practical as with Bin Packing. Numerous criteria are available for selecting the best part, orienting it, and placing it on the resource. One technique used by manual nesters is to search for complementary shapes existing between the profiles of the unplaced parts, and the remaining usable stock. Exact, 'jigsaw puzzle', type fits seldom exist in practice, complicating the judgment of best match. The nesting technician often ignores fine detail and extracts more generic characteristic shapes or features to judge the quality of the actual match between two profiles. The object of the remainder of this manuscript is thus to present a methodology for incorporating these concepts into a comprehensive nesting algorithm.

Interactive Modeling Research Laboratory, Department of Mechanical Engineering, Louisiana State University, Baton Rouge, LA 70803-6413, USA
Paper received: 18 March 1996. Revised: 20 June 1996

PREVIOUS RESEARCH

Reasoning based on part shape has been proposed to varying degrees by several authors to solve the irregular profile allocation problem. In the most elementary form, this represents the matching of profile sides. This is the approach taken by Dagli and Totoglu[5]. Here patterns are allocated onto plates sequentially, with order determined by a set of priorities based on properties such as part area, profile perimeter, and complexity. Starting with the two highest priority parts, their relative locations are determined by pairwise matching each of their sides, selecting the location yielding the smallest minimum enclosing rectangle (MER). The process is then repeated with the next part profile in the prioritized list until all parts are placed, or no parts can be placed on the existing measure. The indiscriminate checking of all possible combinations of sides incurs the largest computational expense associated with the algorithm.

This basic principle of matching sides is also discussed by Prasad[15]. Here, one part is held fixed while the other slides or translates along its boundary as in the no fit polygon (NFP) described by Albano and Sappupo[2]. The relative orientation of the parts is determined by aligning their longest profile edges. An MER is then constructed for each step of this NFP process. As with Dagli, the placement corresponding to the smallest MER is selected as best. Unfortunately, the algorithm is designed for sheet metal stampings, and is limited to problems of only two or three parts.

Rule based or expert systems represent a second category of approach using shape to solve the nesting problem. Cheok and Nee present a three step automatic lay-out process developed for a ship building application[3]. The first stage, called shape processing, produces an approximate description of the part profiles by eliminating fillets, chamfers, and other minor features. These simplified shapes are then classified based on commonly used parts such as floors, rectangular brackets, trapezoidal brackets, etc. In the second stage the classified parts are paired according to predefined arrangements which, based on previous experience, produce 'good' or tightly packed rectangular modules. Stage three then uses a specialized rectangle packing method to nest these modules. Yazu also reports on another rule based method, but for clothes pattern shapes[18]. A large set of specific rules is used for each clothes type, such as men's shirts. However, details concerning these rules and their use are not clearly presented in the discussion.

A familiar and interesting analogy to the nesting problem, that of putting together jigsaw puzzles, was investigated by Freeman[11] and later by Radack and Badler[16]. Radack and Badler's study determined matches based on a novel method for representing the part profiles using a boundary-centred polar encoding. Freeman based the correct placement of parts on comparisons between partial segments or 'chainlets' of the part profiles, chosen such that it was likely there would be only one mate with a chainlet from another piece. Chainlets were produced by dividing the part profiles at 'critical' points defined as inflection points and slope discontinuities in the profile[10]. A high number of chainlets and combinations is possible, therefore a set of rough orientation invariant similarity measures is first used to determine the most likely matches. These are

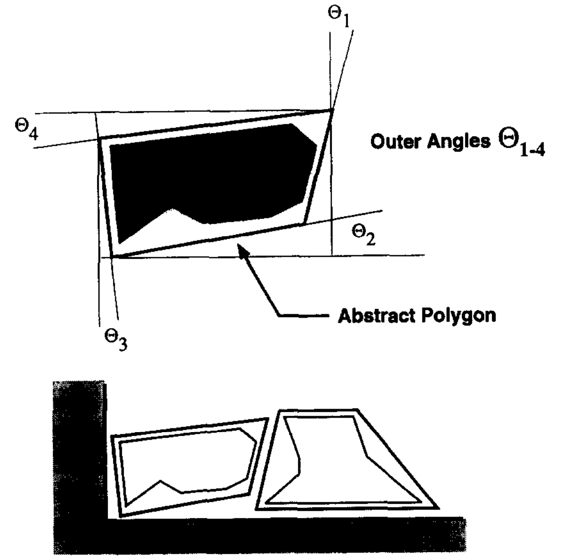


Figure 1 The method proposed by Chung. Each part is represented by an 'abstract polygon'. Parts are placed by matching outer angles

then subjected to a more intensive comparison. The puzzle was assembled by adding individual pieces to a core central piece, growing the solution in an outward direction. In practice several restrictions are placed on the types of puzzles which could be solved. Applying either Freeman's or Radack and Badler's approach is difficult since exact puzzle-like profile matches seldom exist in nesting. Therefore many, but not all of the techniques discussed are of limited use.

Chung and his colleagues apply perhaps the most comprehensive shape based approach to solving the Bin Packing problem, using a series of techniques to orient, place, and pack parts[4]. The primary shape heuristic for placement matches concavities and convexities existing in parts, and defines a best fit adjacent piece for each of four primary orientations (90°, 180°, 270°, 360°). Once a particular part is placed, its 'best' adjacent piece is tested next. Unfortunately, no information is provided on how these best fit pieces are defined. Whenever this logic fails candidates for placement are determined by matching a series of part outer angles (*Figure 1*) with those of the most recently placed part. The outer angles are derived from a polygonal approximation of the profile which must be provided by an experienced technician. The largest of the candidates is selected as the next best fitting part and placements are tweaked further by minor translation and rotation. A quadtree approximation of the parts is used to detect intersections and prevent overlap. The restriction to four basic orientations, although reasonable for nearly rectangular shapes, defines a key limitation when applied to more complex forms requiring additional freedom for effective placement.

The method presented in this manuscript attempts to go beyond simple edge and angle comparisons and provide intelligent heuristics based on more informative geometric characteristics. This must be done while still dealing with shape at a rudimentary level, less case specific than contemporary expert systems. This is made possible by handling the placement ambiguities present in nesting while reasoning about the solution in a way reminiscent of the puzzle problem and a manual nester. This is achieved utilizing the profile features defined in the next section.

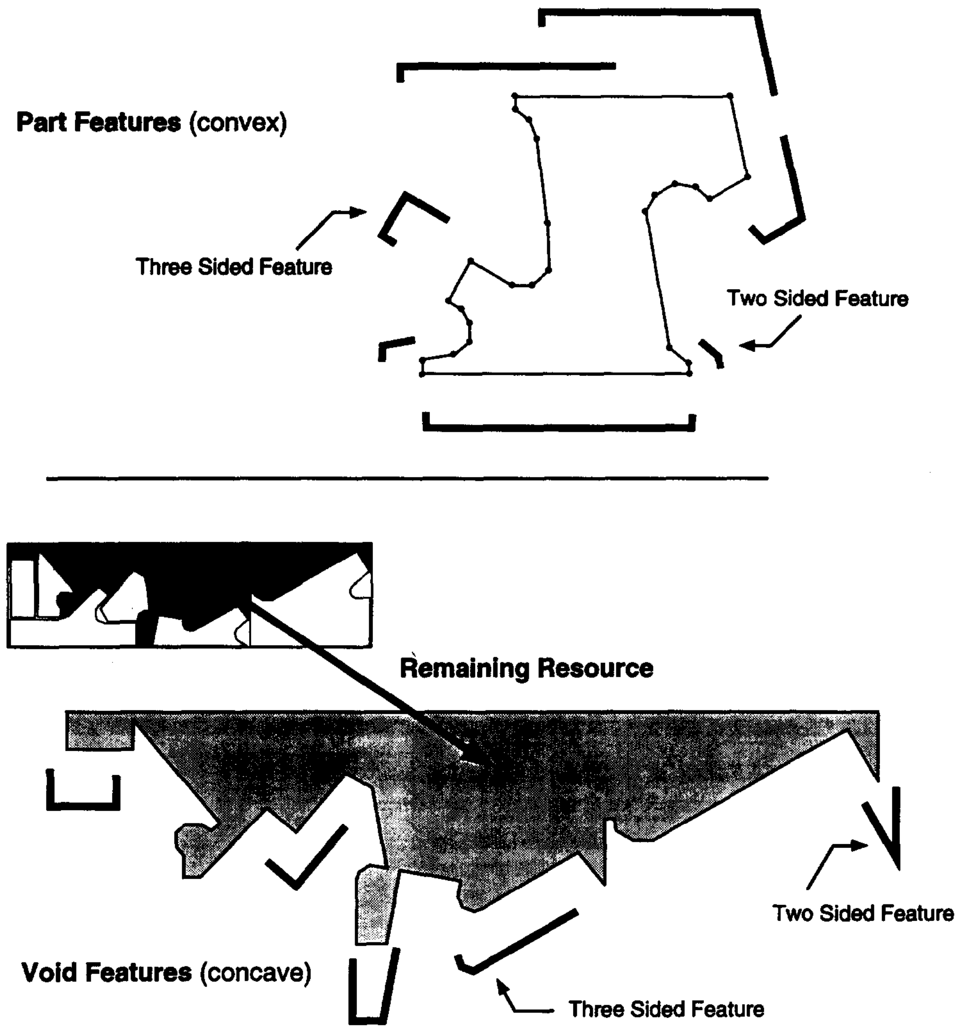


Figure 2 Two and three sided features as consecutive edges on part and void (remaining resource) profiles. Part features are convex, while voids are concave

FEATURES

Recall the objective of this research is to place parts by matching complementary shapes from the placed and unplaced profiles. At their simplest, these shapes can be described by two adjacent edges and their included angle (*Figure 2*). The shape is also distinguished by classifying its vertex as either concave or convex. To extend the concept of shape to true protrusions (peninsulas) and pockets (coves) a third side is added. In most cases, a sufficient description of shape can be achieved using features of two and three sides, since exact matches of shape with three or even two sides are uncommon with irregular profiles. In practice it is only necessary to extract convex features from parts to match with concave features from the remaining stock material, also referred to as the remaining resource or void. Examples of void and part features are shown in *Figure 2*.

Restricted to three sequential edges of the actual profile, the notion of a feature still supplies only a localized characterization of the shape. The general problem of nesting parts of varying size and complexity requires information about shape at various scales. This is illustrated in *Figure 3* where placing small parts requires finer detail but is of little use in positioning larger pieces. To obtain a more global characterization,

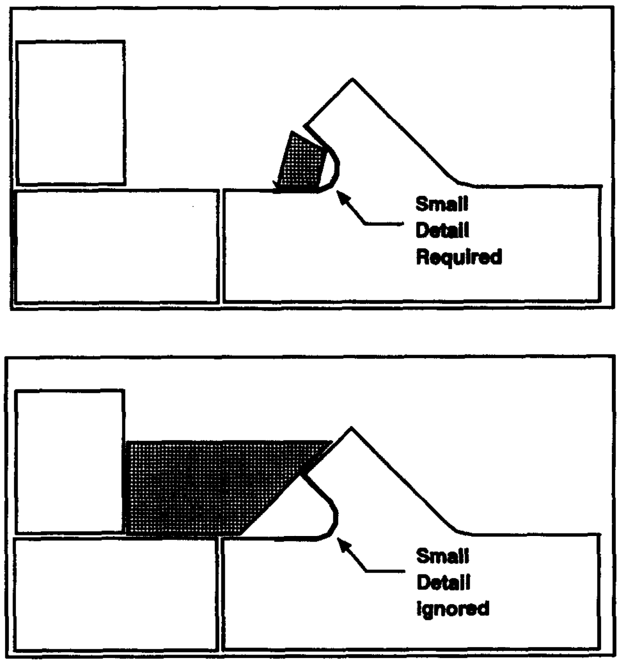


Figure 3 Utility of shape characteristics at various levels of detail during part placement

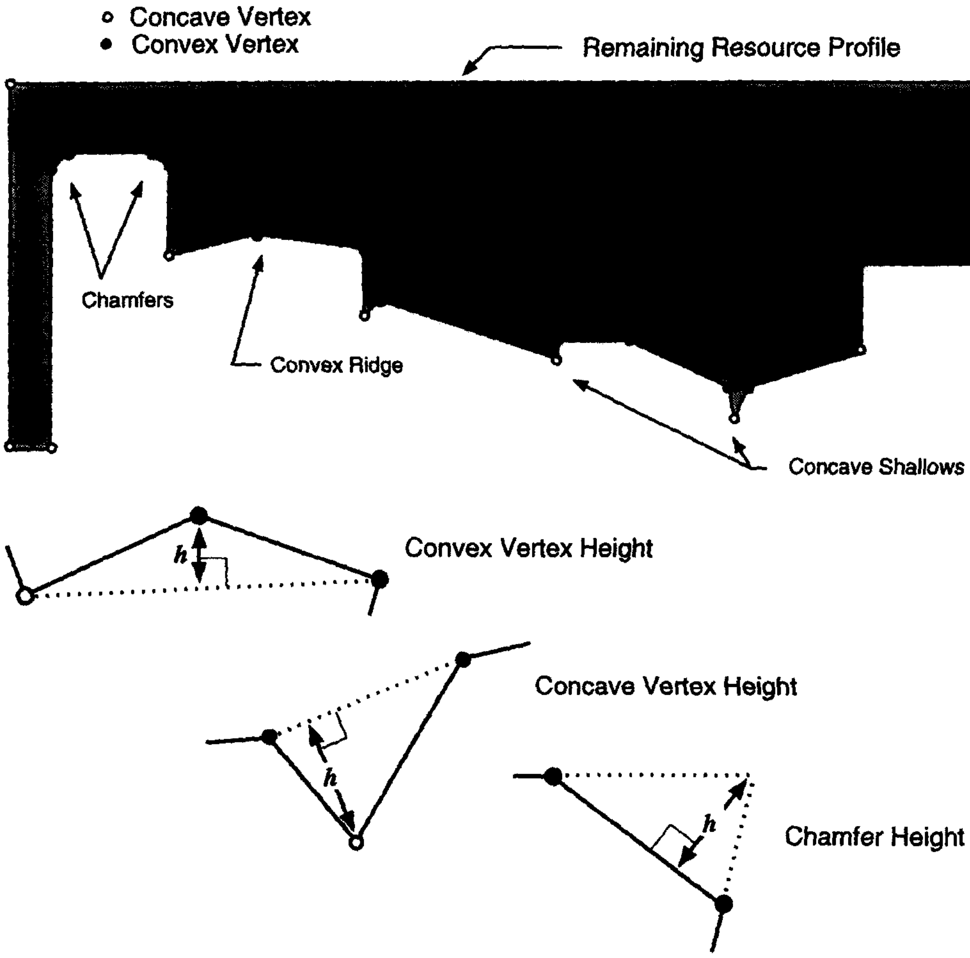


Figure 4 Concave and convex vertices of the remaining profiles and the associated heights of Small Complexity Reduction

simplified profiles which approximate the original shape at varying levels of detail are produced. From these, features are easily extracted using straightforward heuristics which examine each set of three sequential sides and their two included angles. In this way, a hierarchical group of features describing both minuscule and large scale characteristics of a profile can be constructed. The methods used to simplify profiles are detailed in the next section.

PROFILE SIMPLIFICATION

Methods of approximating a part profile with varying degrees of detail have been studied by many[6,9,14,17]. The current method proposes to use features for non-overlapping placement, thus requiring they be extracted from approximations completely inscribing (voids) or circumscribing (parts) the original profile. This is the major impediment to implementing previously reported techniques. Simplification is accomplished by applying a number of different metrics discussed in the following sections. The techniques for processing the remaining resource are outlined first, as part profiles are handled using a sub-set of these same methods.

Simplification of the void is achieved using both Small Complexity Reduction (SCR) and Large Scale Resource Division (LSRD). SCR eliminates inconsequential shallows, ridges, and fillets by examining each profile vertex and its two adjacent sides. Since the effect of such techniques is localized to only a small portion of the profile, the extent of the simplification is limited. LSRD eliminates complexity at a more global level, splitting the void at locations of necking and evaluating the usefulness of the regions based on their shape and size. The remaining resource is simplified when unsuitable areas are eliminated.

Small shallow and ridge irregularities are found and removed in SCR using similar methods. Each vertex of the void profile is classified as either concave (shallows) or convex (ridges) and its associated height determined (*Figure 4*). A candidate for elimination exists wherever this height falls below a tolerance set from experience to be representative of smaller details. Methods for dealing with the various cases that occur are shown in *Figure 5*. Each has in common the elimination of at least one vertex and edge.

SCR examines chamfer irregularities based on the extension of two adjacent convex vertices' edges. The intersection of the two corresponding sides is found, and its distance from the shared edge determined. If this height falls below the ridge/shallow tolerance, a corner is extended as shown in *Figure 5*. Additional issues such as checking the new edges for intersection with the current profile are also involved[12].

The second resource simplification, LSRD, follows SCR and is achieved through splitting the void into several regions at areas where necking occurs. The objective here is to maintain larger useful areas, while eliminating smaller unusable ones where the likelihood

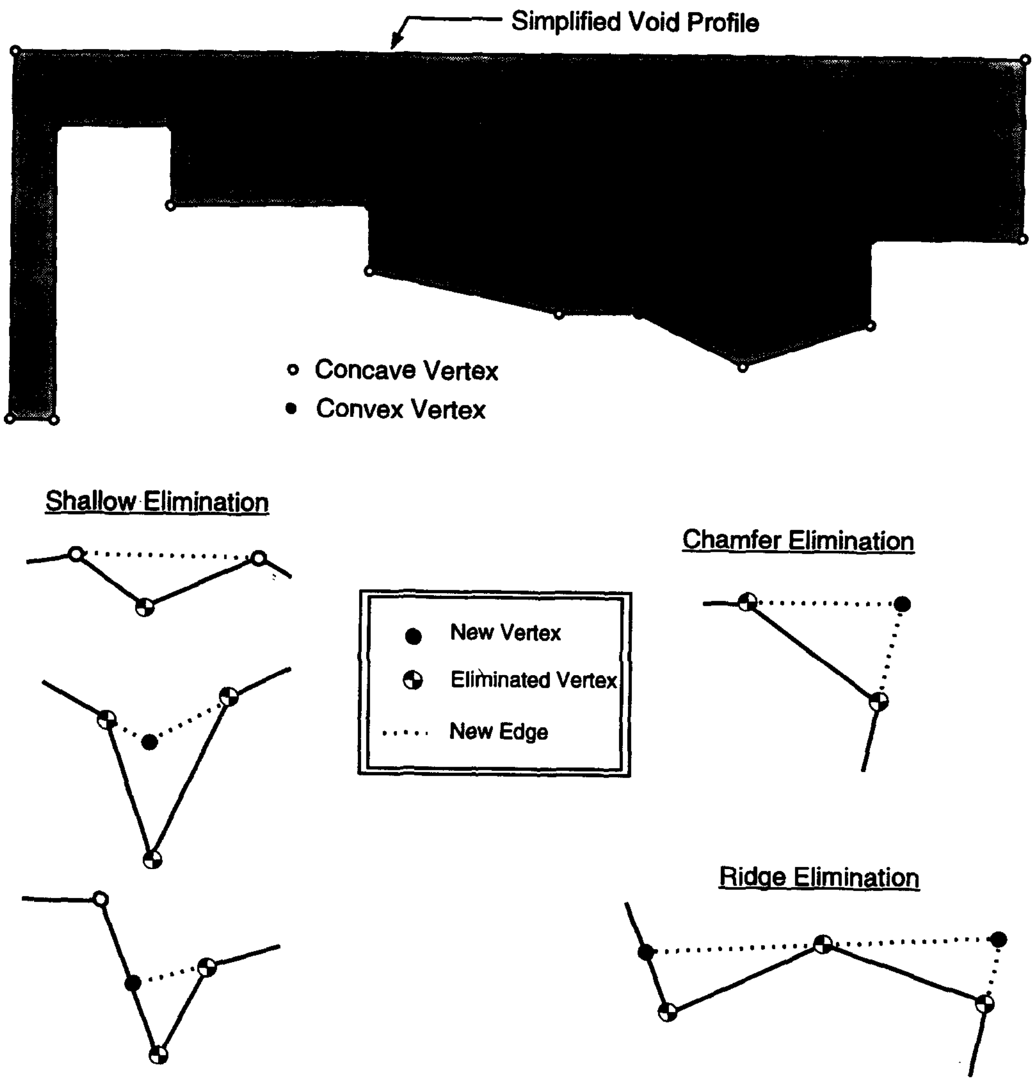


Figure 5 The void profile after SCR processing plus methods used for eliminating shallows, chamfers and ridges

of successful placement of parts is minimal. Simplification is achieved in multiple steps through the decomposition of the void along Dividing Lines (DLs) constructed between each convex vertex and the nearest point on a facing edge. This is graphically depicted in *Figure 6*. At each level three actions may be taken with each of the available sub-regions.

- DIVIDE: Divide the subregion at its shortest DL.
- ELIMINATE: Simplify the remaining resource by completely eliminating the subregion.
- ACCEPT: End processing of the subregion and incorporate it into the simplified description of the remaining resource.

Four criteria determine the action taken with each subregion. *Figure 7* indicates how these are used.

- AREA: True if the area of the subregion is smaller than a set tolerance (*AREA_TOL*).
- CONCAVE: True if all vertices of the subregion are concave.
- SIDE: True if one or none of the edges describing the subregion is a temporary dividing line. Dividing lines whose lengths fall below a set tolerance (*LENGTH_TOL*) are considered temporary.
- MER: True if the shortest side of the MER for the subregion is smaller than a set length tolerance.

The general intent is to eliminate small regions and divide larger ones until only convex areas remain. The MER of the subregion is used to detect long and narrow areas which can be eliminated. Limiting the number of temporary DLs to describe the boundary of a subregion minimizes the potential for regions becoming so small that they are unnecessarily eliminated.

The degree of simplification produced by LSRD is determined by the values of *AREA_TOL* and *LENGTH_TOL*, which in practice are adjusted to reflect the representative size and dimension of the set of remaining parts. To achieve this goal, each profile's characteristic dimensions are represented by the height and width of its MER, while its area is used to indicate size. Parts previously placed and those larger than the current resource are eliminated. The remaining values are then sorted, and averages for the smallest, middle, and largest third tabulated. Three tolerance pairs are generated and when used in the LSRD algorithm produce the different levels of detail seen in *Figure 8*.

Approximations of the original part profiles are produced using the same SCR methods described for

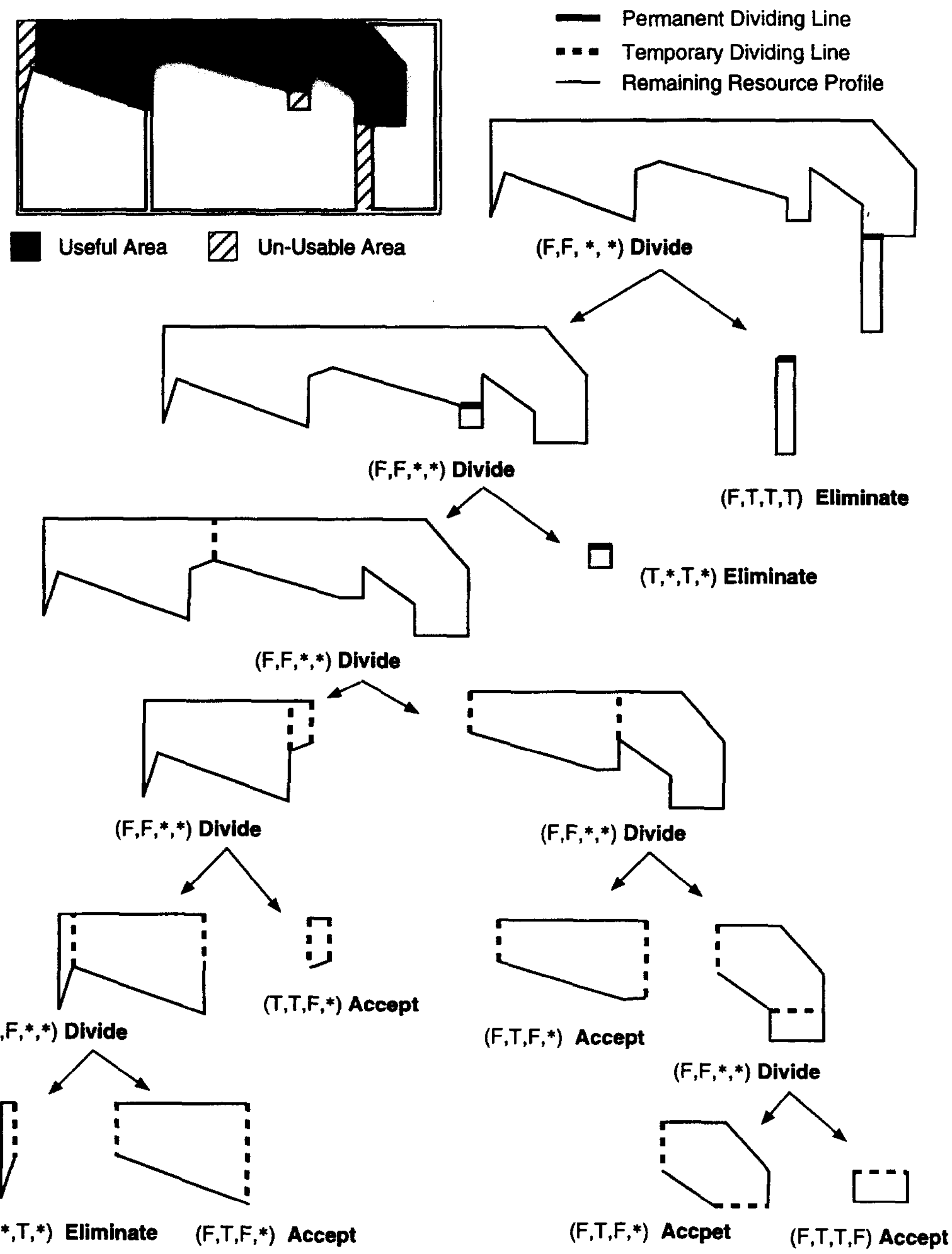


Figure 6 Idealized example of the LSRD process. Quadruples preceding each action represent values of the decision criteria (AREA, CONCAVE, SIDE, MER) for that subregion. An asterisk indicates the criterion is not required to determine the shown action

the remaining resource. Differing levels of detail can be achieved by varying the maximum allowable height of the irregularities (*Figure 9*). As this differs depending on the size of the part, the tolerance is set dynamically as a percent of the profile's MER diagonal.

USE OF FEATURES IN NESTING

Once simplified profiles have been produced, part and void features can be extracted and stored. The geometric information described by these features plays a dual role at each stage of the solution. The first is to limit the placement options available to a workable number of possibilities. Recall that each of the available parts may be placed in an infinite number of locations and orientations. However, analogous to the manual nesting process, the number of choices is reduced by 'matching' only void and part features with the most complementary shape. Once this match has been selected, the features are further used to orient and locate the part within the remaining resource. In this way, both part selection and placement are based on the use of features. Part placement is detailed first as several concepts common to both tasks are best understood in its context.

Placement of a part onto the resource requires establishing both its orientation and location. For each of the many pairwise feature combinations, the current method supplies three potential orientations through alignment of one of the three sides of the part feature, with its corresponding side on the void feature (*Figure 10*). These aligned edges are referred to as the *primary sides* of the match.

Once the primary side of a match is selected, the way in which the two features mesh or fit together is ascertained.

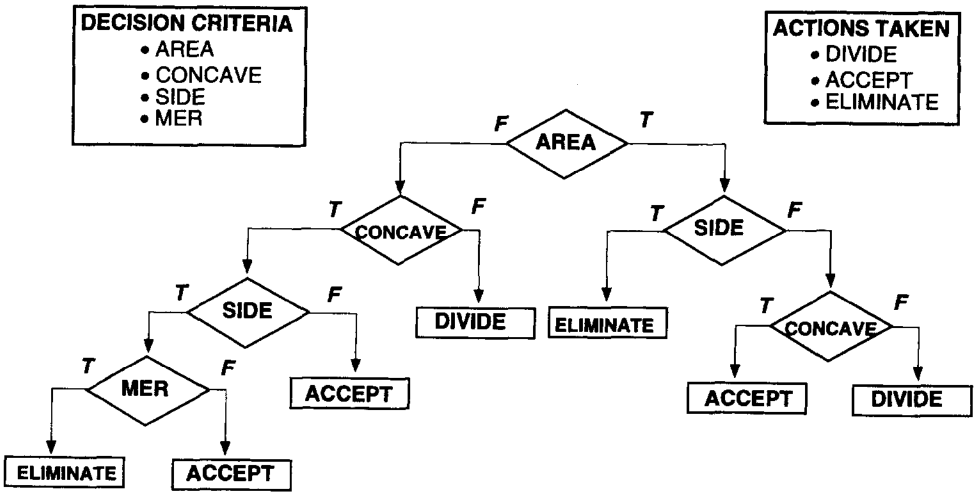


Figure 7 Decision tree showing the action taken for void subregion criteria

Figure 11 shows different fits or 'align types' when associating the middle edge of two features. Each align type characterizes the farthest extent to which a part feature may be pushed into the void feature. The objective is to provide an initial part placement totally contained within the remaining resource and adjacent to a complementary shape. For each align type, a system of linear equations and inequalities represents the necessary and sufficient conditions for that case to occur[12]. A representative example is shown in *Figure 12*. By examining these criteria, the align type for a particular pair of features is established.

Because features are only an approximate representation of the actual profiles, the initial part placement indicated by the align type is only an estimate. For those placements residing totally within the void, further refinement is achieved by translating the part as far as possible in a predefined direction. This *shift direction* is calculated from the angles of the void feature. The align type for a feature pair dictates which of three possible formulations of direction is used (*Figure 13*). Final placement is accomplished by translating or sliding the part along the boundary of the remaining resource until movement opposite to the shift direction is detected.

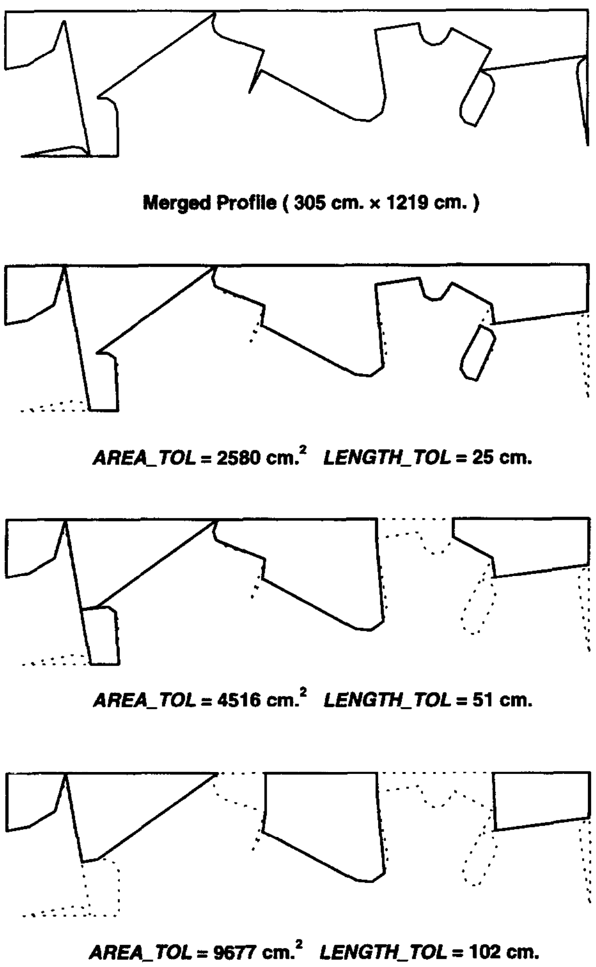


Figure 8 Different levels of detail produced by varying the area and division length tolerances of the void simplification algorithm

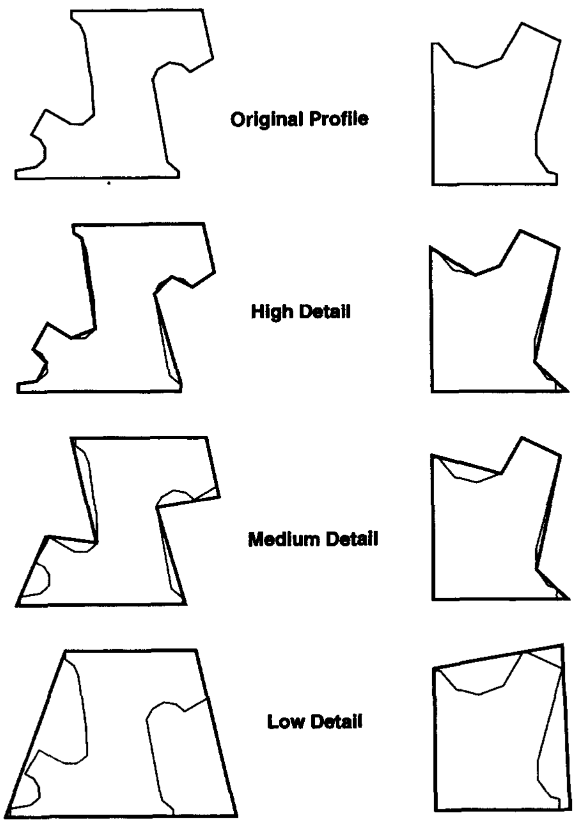


Figure 9 Different levels of detail produced by the part profile simplification algorithm

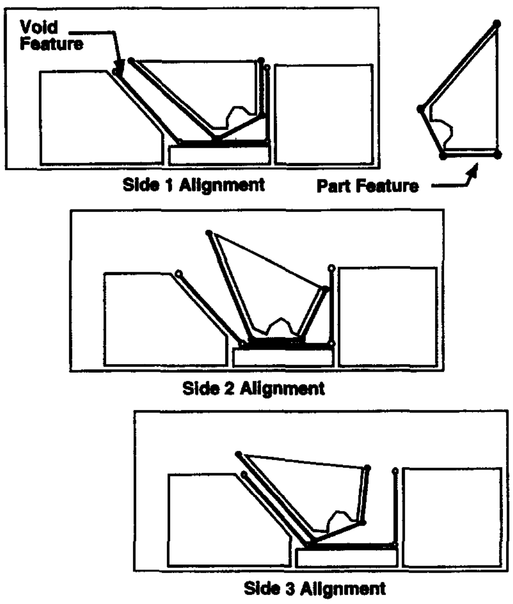


Figure 10 Potential feature–void side alignments

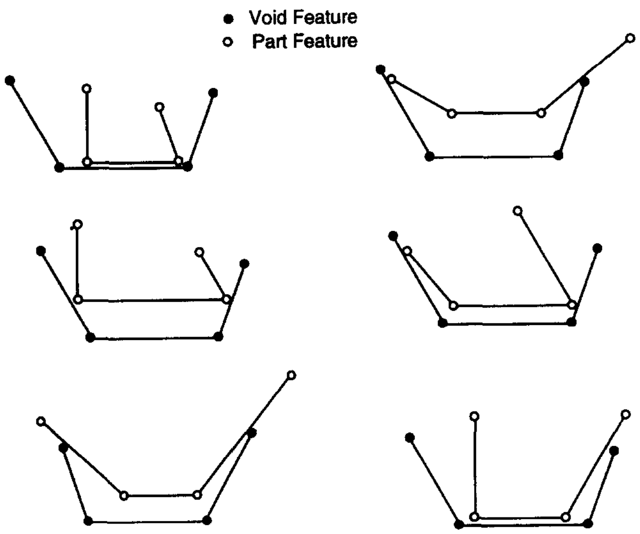
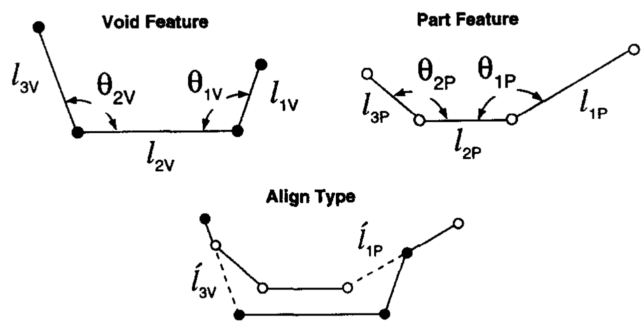


Figure 11 Different fits or *align types* possible between two features with centre edge as the primary side



$$0 = l_{1V}\cos\theta_{1V} - l_{2V} + \acute{l}_{3V}\cos\theta_{2V} - l_{3P}\cos\theta_{2P} + l_{2P} - \acute{l}_{1P}\cos\theta_{1P}$$

$$0 = l_{1V}\sin\theta_{1V} + \acute{l}_{3V}\sin\theta_{2V} - l_{3P}\sin\theta_{2P} + \acute{l}_{1P}\sin\theta_{1P}\max$$

$$TRUE \;\; iff \; \left(0 < \acute{l}_{3V} \le l_{3V}\right) \; AND \; \left(0 < \acute{l}_{1P} \le l_{1P}\right)$$

Figure 12 The system of equations establishing the necessary conditions for an align type to exist. The relation is expressed in terms of the information stored with each feature

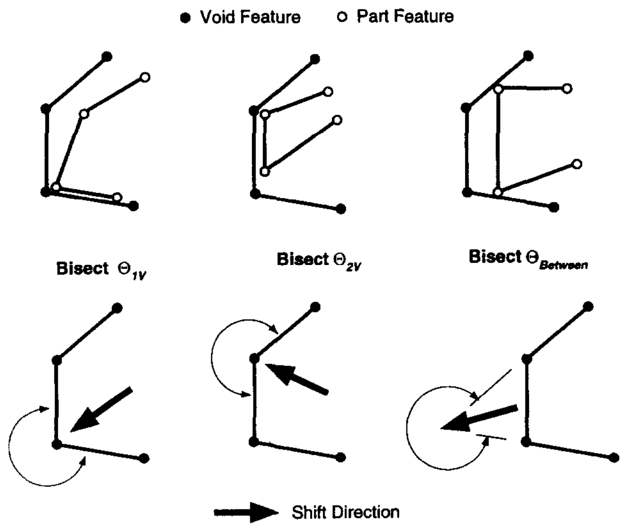


Figure 13 The three possible shift directions determined by the void feature angles and the align type of the match

One consequence of the shape reasoning heuristics is a natural affinity to stack parts along long straight edges and square corners. However, long edges and square corners are typical of stock materials. Experience shows this bias causes placement to proceed unfavourably inward from the outer boundaries of the plate. To overcome this tendency, restraints are placed on the use of those features containing elements of the original stock material border, also referenced as *border features*. When a border feature is involved, the shift direction must fall between two limits, otherwise the match is ignored. For example, parts will favour the upper left-hand corner of a rectangular resource, when border feature shift directions are limited between 90° and 180°. In such an arrangement, placements proceed from the left downward in a more beneficial manner. Additional control is also obtained by varying the acceptable shift direction range between part placements. The characteristics of the solution can be changed considerably using these border feature shift directions.

To summarize, four steps are involved in placing a part (*Figure 14*).

(1) Choose a primary feature side to orient the part with respect to the resource.
(2) Determine the align type for the associated features.
(3) Position the part using the align type and insure it falls totally within the remaining resource.
(4) Shift the part as far as possible within the void and in the direction specified by the align type.

Feature matching

As indicated above, given a matched feature pair it is possible to determine if a valid part placement exists. In practice it is not unusual for each part to have 5–10 features while the remaining resource may have upwards of 20. The large number of combinations make it impractical to examine all possible pairings. As a consequence a quick and computationally efficient method for determining the quality of match between two

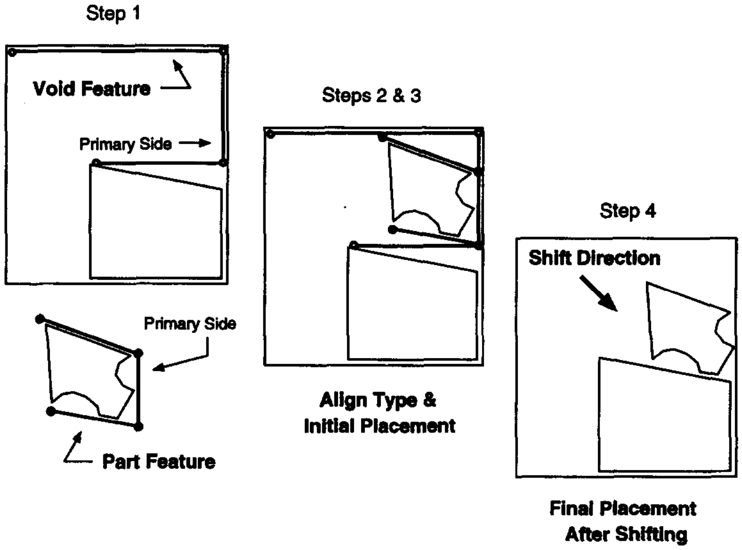


Figure 14 The four steps of part placement: (1) select features and orient part, (2) determine align type, (3) initial placement, (4) part shifting

features is merited. The strategy used is to formulate a single number measure or *matching index* for each possible alignment of a pair.

The matching index is a combination of three basic difference measures between the part and void, using information stored with each feature, namely two included angles and the length of each side (refer to *Figure 12*). For the case where edge one is the primary side, the three values are:

$$X = |(l_{1V} - l_{1P})/l_{1V}|$$
$$X \leqslant \text{primary side tolerance} \quad (1)$$

$$\beta = |\theta_{1V} - \theta_{1P}|$$
$$\beta \leqslant \text{matching angle tolerance} \quad (2)$$

$$Y = |(l_{2V} - l_{2P} \cdot \cos\beta)/l_{2V}|$$
$$Y \leqslant \text{secondary side tolerance} \quad (3)$$

X reflects the normalized length difference between the primary sides of the match; β is the angular difference between the *secondary sides* (i.e. the edges of the match corresponding most closely with the primary side). With alignment of edges one or three the centre edge (*Figure 15*) is secondary, while for centre edge alignment the side producing the smallest value of β is considered secondary. Finally, Y is the normalized difference between the projected length of one secondary side onto the other. Matches falling within tolerances are weighted to produce a single index as follows:

$$\text{MATCH INDEX} = \xi(X) + \phi(\beta) + \psi(Y). \quad (4)$$

The weighting functions $\xi(\ )$, $\phi(\ )$, and $\psi(\ )$ are formulated to assign the desired importance to each variable. For the current implementation the angular comparison β plays the most significant role in determining the closeness of a match, followed by X then Y, respectively.

IMPLEMENTATION

The methods described to this point are integrated into a search algorithm which generates solutions through the iterative placement of parts. In this section, the key steps of this process are first outlined, followed by a more detailed discussion of practical considerations for their implementation.

Search algorithm

Figure 16 shows a flow diagram of the primary steps involved in generating nesting solutions. Part profiles are first preprocessed. At this stage, curves are replaced with piecewise linear approximations, and if necessary for manufacturing, profiles are offset to maintain a specified

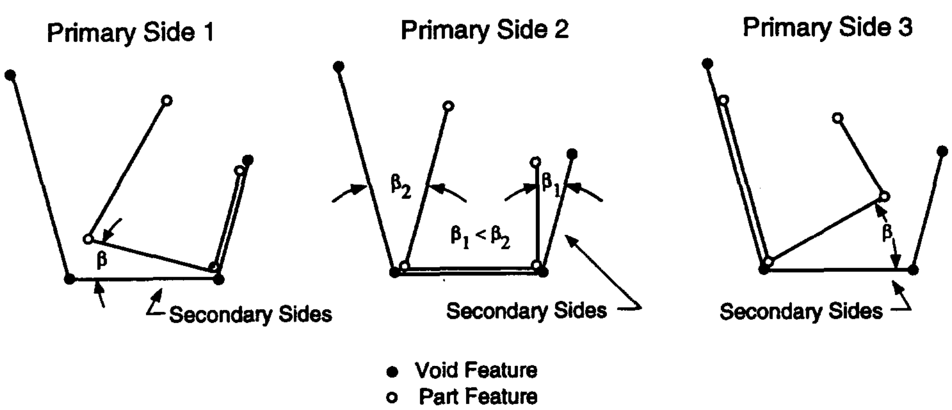


Figure 15 Angle β and secondary sides for the three primary side alignments

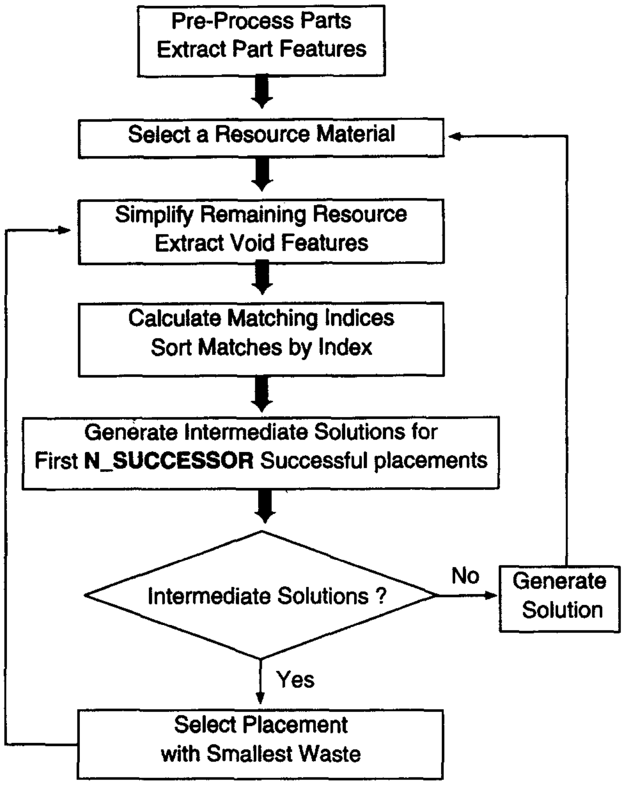


Figure 16 Flow diagram of the primary steps in feature based nesting

distance between parts on the final layout. The primary role of preprocessing however, is to extract part features at varying levels of detail. The geometric feature information is extracted once and stored in a large database which is accessed repeatedly during processing.

Following preprocessing, the stock material for the layout is selected. The algorithm then advances into a large decision loop constituting the majority of work. Each iteration of this loop represents the addition of one part to the final solution. The first step in placing the next part is simplification of the resource remaining from the previous stage. Initially this resource is itself the entire stock material. Once simplifications of the remaining resource at varying levels of detail are produced, void features are easily extracted and stored. Matching indices for all possible feature combinations are then generated from the part and void database. Progressing from most to least favourable indices, the placements indicated by each pairwise match are examined and invalid placements due to overlap eliminated. Each valid placement is called a *successor* of the current solution, and its associated part orientation and location are saved. A preset number of successors ($N\_SUCCESSOR$) is generated and, assuming that valid placements exist, only that successor with the lowest waste value is used for further expansion. The process is then repeated with the remaining resource recorded for this successor. If there are no successors, additional parts cannot be placed on this resource. The current solution is then output and nesting is initiated on a new stock blank. The complete solution is realized when all parts have been placed.

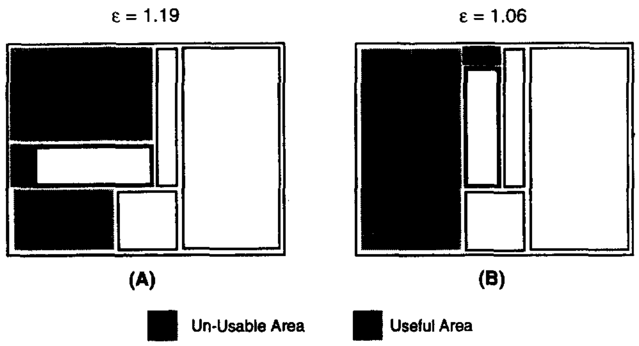


Figure 18 Two placements of the same part generating equal amounts of unusable and useful area. Case A has the higher perimeter penalty

Waste function

The waste function distinguishes between the successors generated for each intermediate solution. It consists of two terms assessing the current waste and predicting the future waste of a solution. Future waste is an estimate of the penalty incurred by allocating all remaining parts, while current waste deals with the actual trim loss on the existing layout. This distinction is similar to that made by Albano and Sappupo[2].

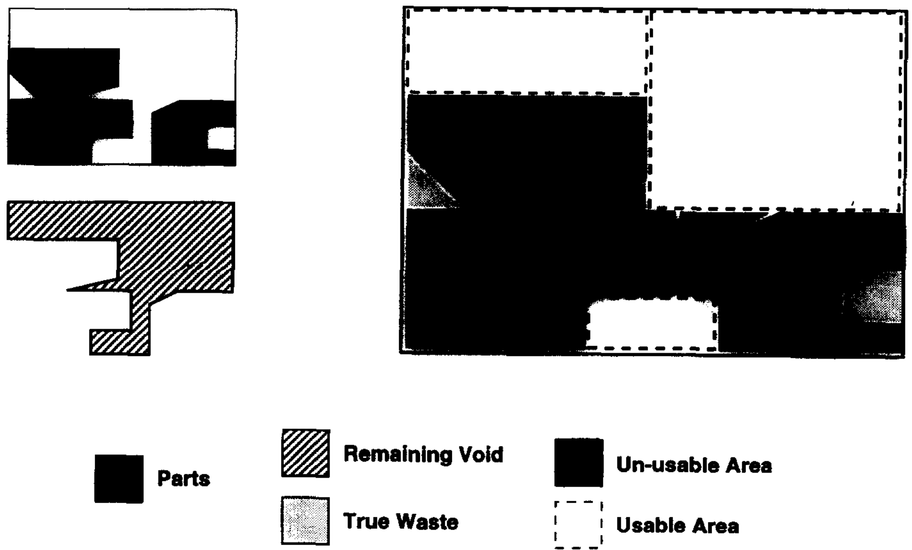


Figure 17 Elements of the waste function

Table 1 Problem set overall characteristics

Problem set	Number of parts	Number of masters	Average normalized area ×100	Total normalized area ×100	Average irregularity	Average concavity	Average complexity	Stock height, cm	Stock width, cm
A	70	50	7.55	528	1.33	1.00	12.2	307	1280
B	109	70	19.48	2124	1.15	1.00	11.0	307	1402
C	206	70	1.84	378	1.18	1.00	6.6	183	914
D	103	60	3.57	368	1.40	1.11	8.3	305	1219
E	295	197	2.41	711	1.05	1.03	6.0	305	1219
F	52	31	5.05	263	1.21	1.12	6.6	305	1219
G	84	49	2.95	247	1.00	1.00	5.5	305	1219
H	56	35	6.37	357	1.39	1.18	12.1	305	1219
I	107	57	6.65	711	1.25	1.08	8.3	305	1219
J	111	50	5.21	578	1.18	1.10	8.2	305	1219
K	37	21	2.85	106	1.08	1.01	5.7	305	1219
L	52	35	7.42	386	1.17	1.05	7.6	305	1219
M	68	32	2.60	177	1.13	1.07	6.6	305	1219

A percentage of each available profile's concave and true area is used to calculate future waste. The region between a part's exact and convex hull profile is its concave area. Once allocated, a part's related future waste is eliminated. This approach favours the early placement of larger and more complex profiles. This is desirable as experience shows such parts are generally more difficult to place.

Current waste consists of two terms, true waste and unusable area. True waste is the unusable area or gaps trapped between parts (*Figure 17*), which are no longer contained in the remaining resource and not accessible for nesting. Unusable areas are the small recesses, notches, chamfers and necks detected by LSRD during void profile simplification. For this use of LSRD, the required area and length tolerances are calculated using only the smallest third of remaining parts. The policy prevents the unnecessary inclusion of waste from areas possibly usable by lesser parts. Although technically contained within the remaining resource, unusable area is considered inaccessible based on the size and shape of the remaining parts. Regions remaining after simplification are referred to as *usable areas*.

The distinction between good and bad placements is often unclear when based solely upon future waste, true waste, and unusable area. For example, *Figure 18* demonstrates two placements of varying quality which generate equal waste measures. For this reason a *perimeter penalty* is calculated to provide additional information for each successor. The formulation is

$$\varepsilon = \frac{2 \cdot \Sigma(A_i)}{\sum\left(\frac{\Phi(A_i)}{P_i} A_i\right) + \left(\frac{\Phi(\Sigma A_i)}{\Sigma(P_i)}\right)\Sigma A_i} \tag{5}$$

where P_i is the perimeter of the ith usable area, A_i is the area of the ith usable area, and $\Phi(x)$ is the returns perimeter of square of area x. The summations are taken over the usable subregions produced during the remaining resource simplification. The ratios $\Phi(A_i)/P_i$ and $\Phi(\Sigma A_i)/\Sigma(P_i)$ are the key terms of the equation. Each estimates the 'squareness' of a region by comparing its perimeter to that of a square of equal area. The first penalizes the subregions individually, while the second examines them as a sum. The overall effect is to favour a single rectangular region over several small irregular ones. Using the perimeter penalty, the total waste is calculated as:

$$\text{Total waste} = \text{Future waste} + 1.5(\text{True waste}) + \varepsilon(\text{Unusable area}) \tag{6}$$

Matching index tolerances and restrictions

During the solution process, the total number of potential matches between part and void features can be very large. For sizable problems, sets of 500 or more part features and 20 or more void features are common. Also recall that for each set pair of void and part features, there are three matches corresponding to the three allowed edge orientations. Such a situation would require the calculation and storage of 30,000 matching indices. Since the majority of these represent invalid placements and only a limited number of successors are produced at each stage of the solution, substantially fewer matching indices need to be calculated. This is accomplished through two restrictions.

A large portion of potential combinations are eliminated by limiting the raw values (X, β, Y) used to calculate the matching index. These tolerances are set dynamically throughout the solution, but are most restrictive initially when the pool of available part features is large. After each stage of the solution, the number of successors generated is examined. If the desired number was produced, the tolerances are tightened. Otherwise they are relaxed to provide more potential matches.

Information about a potential match is also provided by its align type, forming the basis of the second restriction on the pool of produced matching indices. If the values of X, β, and Y all fall below a set tolerance, the align type of the features is established. Only more favourable align types are allowed while all others are eliminated. Determination of the align type also allows for calculation of the shift direction which can then be used to eliminate some matches involving border features.

If no successors can be generated from the set of examined feature matches, all restrictions on align type, border features, and matching tolerances are relaxed. A new set of matching indices is then produced and investigated. This ensures that all possible placements are tested and sometimes permits additional allocations.

Table 2 Summary of results with border restrictions set to 180–360°. Asterisks denote feature based solutions requiring fewer plates than the NFP technique

Problem set	Total CPU time, s			Total plates used			% Usage final plate		
	N_SUCCESSOR		NFP	*N_SUCCESSOR*		NFP	*N_SUCCESSOR*		NFP
	10	20		10	20		10	20	
A	967	1283	1698	9	9	9	40.5	40.1	41.0
B	451	723	993	30	29	30	18.4	65.8*	26.2
C	2665	5406	5899	5	5	5	41.3	36.2	45.7
D	1773	2235	3579	5	5	5	54.4	53.6	50.2
E	2912	5852	8889	9	8	9	1.3	53.6*	4.4
F	240	380	239	4	4	4	13.3	6.4	20.0
G	285	388	320	3	3	3	53.8	46.2	55.7
H	407	821	1296	6	5	6	10.7	68.1*	11.7
I	664	1128	1629	10	10	10	9.0	4.0	38.0
J	1058	1999	3691	7	8	8	78.7*	12.9	17.7
K	73	117	147	2	2	2	13.5	12.4	21.0
L	266	411	394	5	5	5	60.2	60.4	64.9
M	192	397	586	3	3	3	4.6	5.9	11.6

Table 3 Summary of results with border restrictions alternating between 90 and 180°, and 180 and 270°. Asterisks denote feature based solutions requiring fewer plates than the NFP technique

Problem set	Total CPU time, s			Total plates used			% Usage final plate		
	N_SUCCESSOR		NFP	*N_SUCCESSOR*		NFP	*N_SUCCESSOR*		NFP
	10	20		10	20		10	20	
A	459	1000	1698	9	9	9	40.5	40.5	41.0
B	462	674	993	29	29	30	65.8*	65.8*	26.2
C	2062	3040	5899	5	5	5	31.2	34.1	45.7
D	977	1880	3579	5	5	5	54.6	37.1	50.2
E	1337	3339	8889	8	8	9	56.2*	45.8*	4.4
F	139	225	239	4	4	4	9.7	6.4	20.0
G	159	323	320	3	3	3	51.1	47.9	55.7
H	279	440	1296	6	5	6	2.7	61.9*	11.7
I	453	814	1629	10	9	10	10.7	62.3*	38.0
J	603	1999	3691	8	8	8	12.2	12.9	17.7
K	43	117	147	2	2	2	12.6	12.4	21.0
L	185	411	394	5	5	5	60.2	60.4	64.9
M	173	397	586	3	3	3	10.1	5.9	11.6

Table 4 Summary of CPU time per part results for all cases

Problem set	CPU time per part, s					Number of parts
	N_SUCCESSOR = 10		*N_SUCCESSOR* = 20		NFP	
	180–360°	90–180° ⇔ 180–270°	180–360°	90–180° ⇔ 180–270°		
A	13.81	6.56	18.33	14.29	24.26	70
B	4.14	4.24	6.63	6.18	9.11	109
C	12.94	10.01	26.24	14.76	28.64	206
D	17.21	9.49	21.70	18.25	34.77	103
E	9.87	4.53	19.84	11.32	30.13	295
F	4.62	2.67	7.31	4.33	4.60	52
G	3.39	1.90	4.62	3.85	3.81	84
H	7.27	4.98	14.66	7.86	23.14	56
I	6.21	4.23	10.54	7.61	15.22	107
J	9.53	5.43	18.01	18.01	33.25	111
K	1.97	1.16	3.16	3.16	3.97	37
L	5.12	3.56	7.90	7.90	7.58	52
M	2.82	2.54	5.84	5.84	8.62	68
Totals	8.85	5.43	15.66	10.86	21.75	1350

RESULTS

The solution method presented here has been implemented in the C programming language and is currently being run on a VAX 4000-300 server under the VMS operating system. An industrial marine fabricator provided a diverse set of 42 actual problems to evaluate the robustness of the approach. Each problem included a data file defining the geometry of all unique or *master* parts and a list of their instances. Part decriptions consisted of straight lines and circular segments which were approximated with the appropriate inscribed (voids) or circumscribed (parts) chords. Each part is simplified using a modified convex hull approach[13] to reduce the

Table 5 Individual plate results for problem set C (*Figures 19–21*). *N_SUCCESSOR* equals 20

Plate number	CPU time per part			Number of parts			% Trim loss		
	180–360°	90–180° ⇔ 180–270°	NFP	180–360°	90–180° ⇔ 180–270°	NFP	180–360°	90–180° ⇔ 180–270°	NFP
1	29.0	12.4	45.8	53	57	64	20.29	21.74	23.91
2	16.6	27.2	31.0	49	57	66	18.12	21.74	26.81
3	38.2	8.8	26.7	56	51	26	28.99	26.81	34.06
4	21.4	9.2	6.5	33	27	25	34.78	29.71	26.81
5	14.3	6.1	2.6	15	14	25	63.77	65.94	54.35

Table 6 Individual plate results for problem set H (*Figures 22, 23*). *N_SUCCESSOR* equals 20 with border restrictions alternating between 90–180°, and 180–270°

Plate number	CPU time per part		Number of parts		% Trim loss	
	Feature based	NFP	Feature based	NFP	Feature based	NFP
1	8.0	30.9	15	12	31.37	28.98
2	6.0	33.0	10	14	35.73	37.47
3	10.8	24.5	12	11	26.36	37.15
4	6.0	10.7	10	11	32.68	35.62
5	7.8	12.6	9	6	38.13	36.71
6		0.3		2		88.34

computational expense and is also offset a prescribed amount to account for flame-cutting fabrication. Original part boundaries are plotted in the solution layouts. Internal voids are extracted from larger parts and nested first. The remaining profiles are then allocated to stock standard rectangular plates selected successively from a prioritized list of available materials provided for each problem by industry. The voids and any associated nests are then reinserted onto their parent parts when generating output files.

To help quantify the characteristics of each problem a series of measures was developed. In addition to total part count and number of masters, four measures describing the profiles contained in the problem were defined. These are:

$$\text{Normalized area} = \frac{\text{Area of part profile}}{\text{Area of stock resource}}$$

$$\text{Irregularity} = \frac{\text{Area of part MER}}{\text{Area of part profile}}$$

$$\text{Concavity} = \frac{\text{Area of part convex hull}}{\text{Area of part profile}}$$

Complexity = Number of vertices in profile description

Normalized area gauges the size of a part relative to the selected resource. Irregularity measures the non-rectangularity of a profile[1] while concavity indicates divergence from its convex hull. Complexity refers to the number of vertices in the simplified profile used for processing. A distribution of these measures across the entire part set[12] provides greater understanding of the problem characteristics; however, due to space limitations only average values are presented.

Several runs of the algorithm were conducted to evaluate two key parameters which affect the feature based results, the number of successors and border shift direction restrictions. Two values for each parameter were chosen and the four possible permutations investigated. The two cases for shift direction correspond to a constant range of 180–360° and a case alternating between the ranges of 90 and 180°, 180 and 270°. The number of successors was set to 10 and 20. Experience shows these values produce the most favourable results for rectangular resources.

To reduce the CPU time and analysis overhead, and demonstrate the potential gains of varying the involved parameters, a meaningful and manageable subset of the original 42 industry problems was used to conduct the study. *Table 1* shows the characteristics of these problems. *Tables 2* and *3* summarize the results from the four cases run, including total CPU time and the number of stock plates used. Results for a No Fit Polygon[2] implementation[13] are shown for comparison. For the majority of problems, cumulative waste is identical in each of the four cases, since nesting the total part set requires the same number of stock plates. A better indication of quality is obtained by examining the percent usage for the last plate (*Tables 2* and *3*). A lower value here indicates fewer profiles on the final plate, thus implying better initial packing. Also, more material remains for nesting of additional parts should they exist. The feature based method is also conveniently formulated to nest on large irregular sections of trim waste, if they are retained for later use.

Several observations can be made from *Table 4*, which shows normalized CPU times for all cases of the study. Computation times for the feature based solutions are lower than those of the NFP technique. When using an NFP approach, a tentative placement for *each* of the remaining available parts must be generated at every stage of the solution. Thus in theory, CPU time increases quadratically as a function of the total number of profiles nested[12]. Over the course of a solution, a decrease in CPU time from plate to plate is evident, as a decreasing number of parts is available. This trend is seen in *Tables 5–7* where CPU times per part are shown for each plate in the NFP solution. For the shape based heuristic an increase in parts corresponds to an increased number of matching indices. Calculating indices accounts for a small percentage (5–10%) of total CPU

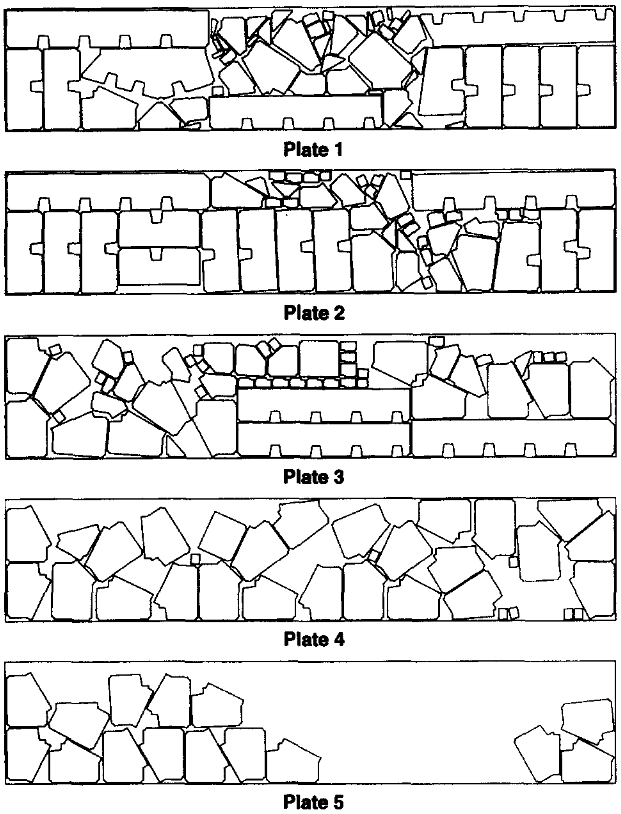


Figure 19 Feature based solution to problem set C. *N_SUCCESSOR* is 20 with border restrictions set to 180–360°

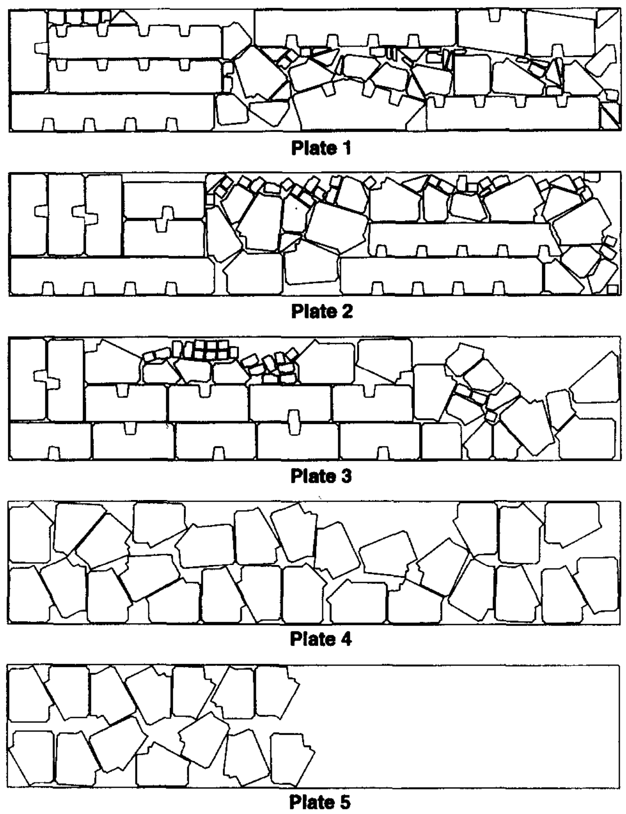


Figure 20 Feature based solution to problem set C. *N_SUCCESSOR* is 20 with border restrictions alternating between 90 and 180°, and 180 and 270°

Table 7 Individual plate results for problem set I (*Figures 24–26*). *N_SUCCESSOR* equals 20

Plate number	CPU time per part 180–360°	90–180° ⇔ 180–270°	NFP	Number of parts 180–360°	90–180° ⇔ 180–270°	NFP	% Trim loss 180–360°	90–180° ⇔ 180–270°	NFP
1	12.8	7.7	36.7	13	11	15	28.24	35.49	22.75
2	9.4	7.6	9.9	18	13	9	21.37	19.90	20.29
3	11.9	6.5	14.2	11	11	10	28.24	20.20	25.59
4	8.1	8.8	32.6	11	12	7	24.71	22.55	39.02
5	8.1	8.3	21.5	14	12	9	20.88	21.96	25.20
6	15.6	7.4	25.1	9	16	9	29.71	17.65	31.08
7	10.3	6.2	8.4	9	10	11	22.75	19.12	31.47
8	10.4	8.9	4.9	10	9	15	22.55	28.82	30.69
9	11.2	7.2	3.0	11	13	11	29.22	37.75	35.49
10	1.4		0.1	1		11	95.98		61.96

time. For the current application, evaluation of the successor waste functions represents the largest portion of CPU time (60–80%). However, since at each stage these calculations are capped by the total number of allowed successors (*N_SUCCESSOR*), solution times are less adversely effected by the number of parts. A roughly linear increase in CPU time can be seen in *Table 4* as the number of successors varies from 10 to 20.

Interestingly, the increase in CPU time associated with more successors does not always improve the solution. Predicting the future effects of a placement based solely upon its remaining resource is often difficult. As a result, generation of additional successors will occasionally allow the waste function to select and expand a less favourable solution. Conversely, reducing successors to too small a number will also adversely affect the solution. Since the matching index only estimates the quality of fit between the part and void, there is no guarantee that the initial placements indicated by it will be the best. Both of these trends can be seen in *Tables 2* and *3*.

The second trend seen in the parameter study data was smaller CPU times for cases where border shift restrictions alternated between 90–180°, and 180–270°. This result can be explained in part by the effect of the restrictions on the formation of the remaining resource. As seen in *Figure 19*, a single range of 180–360° forces parts to migrate toward the lower corners of the stock. By contrast, the alternating case causes parts to favour the left side of the plate (*Figure 20*). In the former situation, the leading edge of the solution tends to extend along the length (longer side) of the plate, while for the later it spans the height (shorter side). Thus remaining voids for the 180–360° case are generally more complex and require larger CPU times for evaluation of their waste functions.

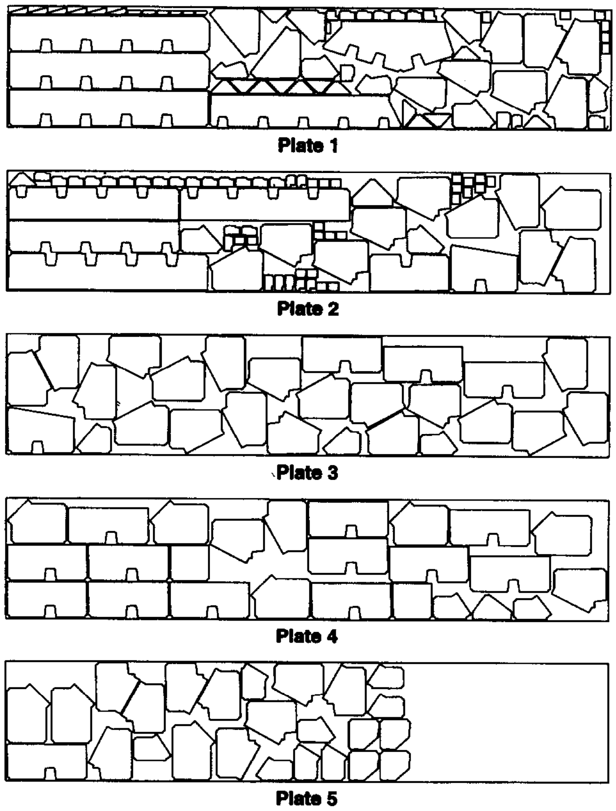


Figure 21 NFP based solution to problem set C

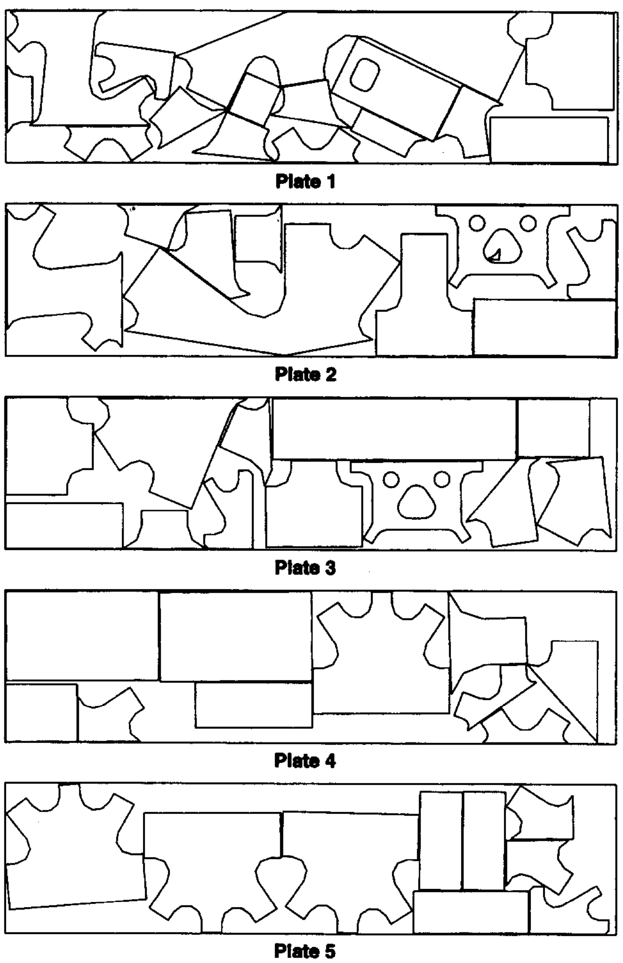


Figure 22 Feature based solution to problem set H. *N_SUCCESSOR* is 20 with border restrictions alternating between 90–180°, and 90–180°

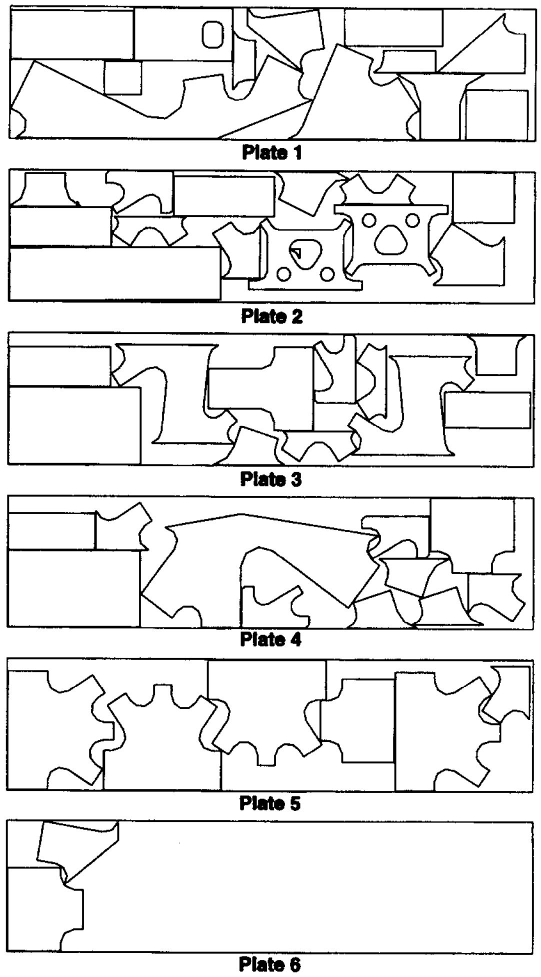


Figure 23 NFP based solution to problem set H

Based on the results of the parameter study a complete run of the 42 industry problems was conducted with alternating shift restrictions and 20 successors as above. Total CPU time to place the 3,014 parts was 8.06 h for the feature based method, while the NFP technique required 11.46. For 35 of the problems, both methods require the same number of stock plates. In six of the remaining seven problems, the feature based technique required one less resource. Representative examples are plotted in *Figures 22–26*. The NFP technique used fewer stock plates (1) in only one of the 42 problems. Material usage on final plates was also investigated. For the comparison if either method required one less resource, percent usage on the final plate for that technique was considered zero. This is a conservative measure since it ignores the available material remaining on that solution's true last plate. Solutions requiring one plate were ignored, as material usage is identical for both methods. Using these analysis criteria an average material savings of 4 percent is observed on the final plate of feature based solutions.

FUTURE WORK AND CONCLUSION

A comprehensive inspection of the layouts generated by the feature based technique indicates nests of comparable waste which are most often superior to those of the

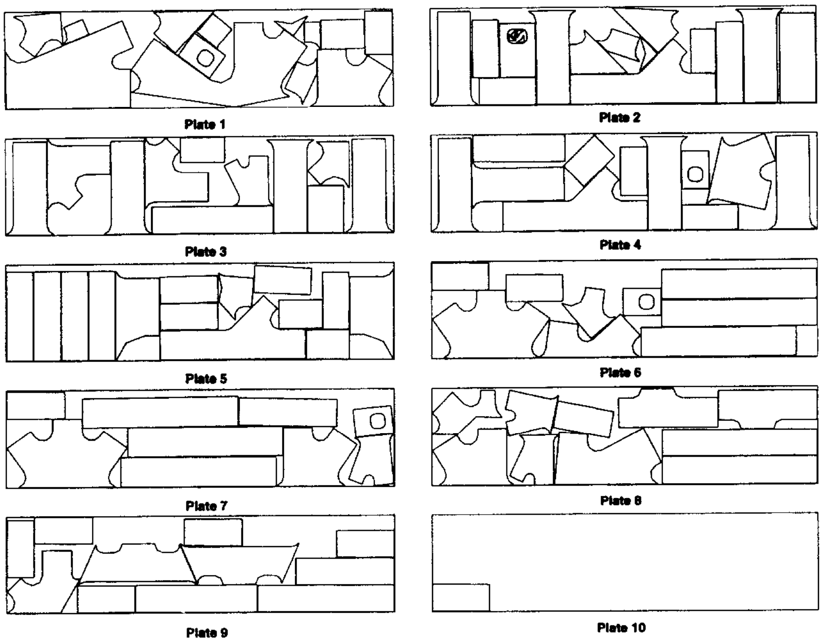


Figure 24 Feature based solution to problem set I. *N_SUCCESSOR* is 20 with border restrictions set to 180–360°

NFP method. Feature based results often contain individual nests with exceptionally low waste values. However, evaluation of the different runs through examination of individual plates is often misleading, as the parts found on each plate differ from solution to solution. Early gains made through the allocation of small or easily packed parts are often lost on later plates when only large and highly irregular profiles remain. Such issues complicate the appraisal of any nesting technique's merits. However, based on results from the total run of 42 problems, the relative merits of this technique for a diverse group of problems are evident. On average, solutions requiring fewer stock materials were produced at less computational expense.

In many cases the nests produced can be improved through manual alteration, however the waste recovered by such operations is often minimal. Any gains made through manual interaction must be weighed against the increased costs in man hours. In industry, nesting involves the optimization of both material and labour used, as well as other issues including cutting equipment, stock availability and scheduling. If the advantages of manual layout are merited, productivity could be gained by incorporating the proposed techniques into a hybrid manual/automatic system. In this application, feature matching heuristics are employed to quickly provide a palate of suggested placements. The user can then select and alter these or if necessary generate his own placements. A more detailed outline of such a system is presented by Albano[1].

One issue deserving additional study is determination of the best successor for expansion. Although

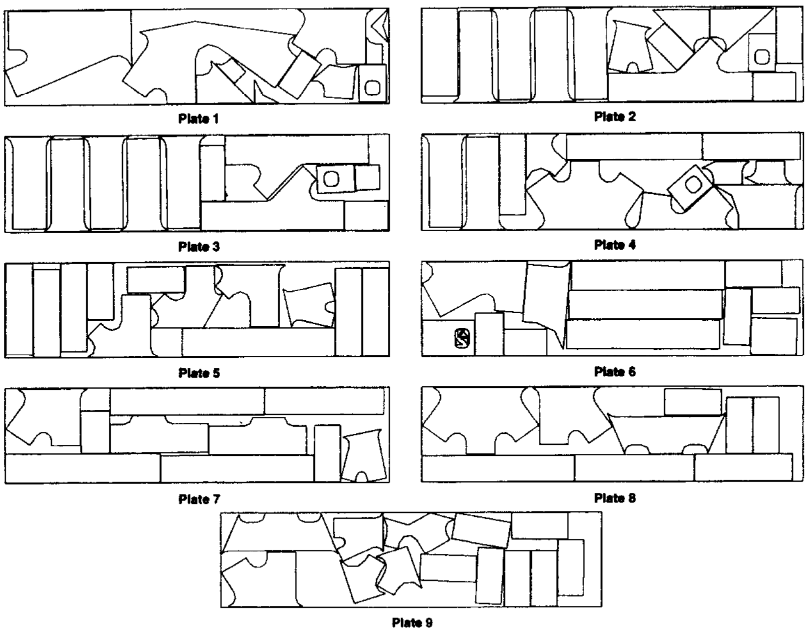


Figure 25 Feature based solution to problem set I. *N_SUCCESSOR* is 20 with border restrictions alternating between 90–180°, and 180–270°

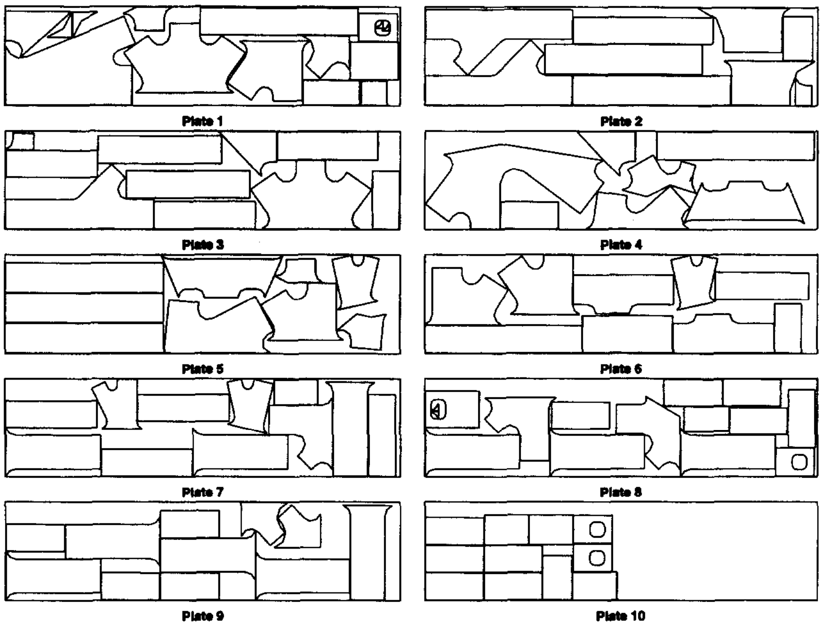


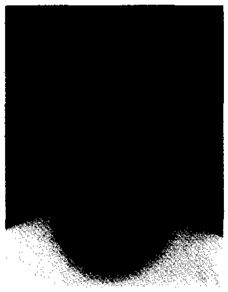
Figure 26 NFP based solution to problem set I

information from the set of remaining parts is used to dynamically control the waste function, it is still inherently limited to examining only the current solution. Unfortunately, the detrimental effect of certain placements can only be seen in successive steps. One solution to such problems is to adopt a less deterministic search algorithm which explores several potential paths at each stage of the solution process. The extra CPU time required to generate the additional placements for such a technique, might be offset by the need for a less rigorous waste function.

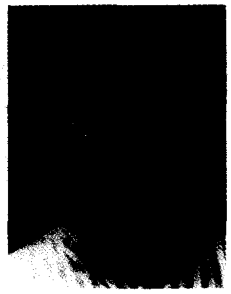
The allocation of irregular shapes is a strongly domain dependent problem. As such, it is difficult to develop generic solution strategies which are optimal when applied to all cases. To varying degrees, expert systems overcome this issue by addressing only specific problem types. As a consequence, their solutions can be effective but limited in application. Unfortunately, industries such as ship and offshore oil platform fabrication generate part sets which are very diverse in structure. For this reason, the method presented deals with profile allocation at the fundamental level of shape. However, ignoring the overall characteristics of the problem set does hinder generating the best results possible. One avenue for future research, is the development of intelligent systems capable of qualifying the nature of problems while processing the nests. Additional shape reasoning will provide useful tools for accomplishing this task. Equipped with this knowledge, the proper application of both generic and rule based techniques should be possible.

REFERENCES

1. Albano, A., A method to improve two-dimensional layout. *Computer-Aided Design*, 1977, **9**(1), 48–52.
2. Albano, A. and Sapuppo, G., Optimal allocation of two-dimensional irregular shapes using heuristic search methods. *IEEE Trans. Systems Management & Cybernetics*, 1980, **SMC-10**, 242–248.
3. Cheok, B. and Nee, Y., Algorithms for nesting and ship/offshore structural plates. *Advances in Design Automation*, 1991, **DE32**(2), 221–226.
4. Chung, J. and Hillman, D. J., Object oriented nesting system on two-dimensional highly irregular resources. *Intelligent Robots and Computer Vision VIII: Systems and Applications SPIE*, 1989, **1193**, 54–63.
5. Dagli, C. H. and Totoglu, M. Y., A computer package for solving cutting stock problems. In *7th International Conference on Production Research*, Windsor, Ontario, Canada, Vol. 1, 1983, pp. 480–486.
6. Davis, L. S., Understanding shape: Angles and sides. *IEEE Trans. Computers*, 1977, **26**(3), 236–242.
7. Dyckhoff, H., A typology of cutting and packing problems. *European Journal of Operational Research*, 1990, **44**(2), 145–159.
8. Dyckhoff, H. and Finke, U., *Cutting and Packing in Production and Distribution: A Typology and Bibliography*. Physica-Verlag, Heidelberg, 1992.
9. Freeman, H., Computer processing of line-drawing images. *Computing Surveys*, 1974, **6**(1), 57–97.
10. Freeman, H., Shape description via the use of critical points. *Pattern Recognition*, 1978, **10**, 159–166.
11. Freeman, H. and Garder, L., Apictorial jigsaw puzzles: The computer solution of a problem in pattern recognition. *IEEE Trans. Electronic Computers*, 1964, **13**, 118–127.
12. Lamousin, H., Allocation of two-dimensional shapes using a shape reasoning heuristic. Ph.D. Thesis, Department of Mechanical Engineering, Louisiana State University, USA, 1996.
13. Lamousin, H., Waggenspack, W. and Dobson, G., Nesting of complex 2-D parts within irregular boundaries. *ASME Manufacturing, Science & Engineering*, 1996, **118**(4), 615–622.
14. Pavlidis, T. and Horowitz, S. L., Segmentation of plane curves. *IEEE Trans. Computers*, 1974, **23**(8), 860–870.
15. Prasad, Y., A set of heuristic algorithms for optimal nesting of two-dimensional irregularly shaped sheet-metal blanks. *Computers in Industry*, 1994, **24**, 55–70.
16. Radack, G. M. and Badler, N. I., Jigsaw puzzle matching using a boundary-centered polar encoding. *Computer Graphics and Image Processing*, 1982, **19**, 1–17.
17. Rosenfeld, A. and Johnston, E., Angle detection on digital curves. *IEEE Trans. Computers*, 1973, **C-22**, 875–878.
18. Yuzu, C., Lujun, L., Wang, W. and Jianwen, S., An expert system for automatic allocation of 2D irregular shapes. In *Proc. Expert Systems in Computer Aided Design Conference*. International Federation for Information Processing, 1987, pp. 407–423.



Henry J Lamousin holds a BS in mechanical engineering from the University of New Orleans and a PhD in mechanical engineering from Louisiana State University. His research interests include computer-aided engineering, scientific visualization and geometric modelling.



Dr Warren N Waggenspack, Jr, holds a joint appointment as Associate Professor of Mechanical Engineering and System Sciences at Louisiana State University. He has been actively involved in research and teaching in computer-aided geometric design and computer graphics since 1981. Dr Waggenspack's continuing research interests include developing effective freeform curve and surface formulations for real-time geometric design, modification and display, general mechanical design and biomedical applications of geometric modelling.