Design and Manufacturing of Sheet Metal Parts: Using Features to Aid Process Planning and Resolve Manufacturability Problems

Cheng-Hua Wang and David A. Bourne

The Robotics Institute

Carnegie Mellon University

Pittsburgh, PA 15213

USA

Robotics and Computer-Integrated Manufacturing, Vol. 13, No. 3, pp. 281-294, 1997

ABSTRACT

This paper describes an integrated system for the design and production of sheet metal parts. We have identified some important features for the sheet metal bending process. These features are automatically generated as the design progresses. After the designs are complete, our automatic process planning system uses the features and generates new ones to aid the production of plans with near-minimum manufacturing costs. Finally, these plans are used to produce parts on an automatic bending system.

Once a plan is generated, it can be used to manufacture the part, and to provide feedback to design and other factory systems. The application of features and the potential feature interaction problems are discussed. Several key manufacturing problems are also considered and the result of planning is used to resolve these problems. By solving these feature interaction problems and often practical manufacturing issues, we are able to plan and manufacture the majority of the parts we have tested under one hour after the flat patterns are prepared.

Keywords: computer-aided process planning (CAPP), sheet-metal design, sheet-metal fabrication, features, feature interaction

1. INTRODUCTION

The sheet metal industry has focused on the automation of punching, shearing and nesting processes for sheet metal parts [10], and only few researchers have investigated the bending operation: Inui and Kimura [9] related product models with features and processes for bending operation; Zussman and Horsch [16] proposed a motion planning approach for robot-assisted multiple-bent parts based on configuration-space and potential filed. de Vin et al. [4] developed a sheet metal CAPP system called PART-S which integrates cutting, nesting, bending and welding processes. Bourne [2] developed an automatic bending process planner to generate plans with near-minimum manufacturing costs. This system consists of several sub-systems that work cooperatively to find the bending plan.

The representation of a sheet metal part should provide sufficient information to make a complete production plan. As a first step, the design must be unambiguous; it should represent one and exactly one part, in either flat pattern or final shape. Secondly, the design must be complete so all the information required to recognize a correct part is present, such as tolerances. Finally, additional information helps identify aspects of the design such as known features. While this last item is not necessary, it can make an extremely difficult planning task relatively easy. We have identified several important features for bending process. These features are later used to help prepare the process plans.

Two approaches have been developed to identify features in a design: feature recognition [6] and design-with-features [5]. The feature recognition approach searches the geometric description of a part for known patterns and then labels the geometry subsets with a feature name. The design-with-features approach allows the designer to directly label the geometry. This has been used in various ways. For example, simulated manufacturing operations have been used to design a part, and then those operation names have been attached to the geometry, so the process planner can relate them to the actual processes [1,11]. Cutkosky et al. [3] have observed that the product development cycle can be reduced by awareness of manufacturing processes during the design process and that the features often carry these relationships.

The process planning for bending sheet metal is considered difficult because the task is very sensitive to small changes in the geometry. A small variation in the part geometry can result in completely different bending sequences. Therefore, using the variant approach to generate the bending process plans may not work even though it has already been tried in the industry. Both de Vin [4] and Bourne [2] used the generative approach to search for the bending process plans. However, the former uses a backward reasoning approach (from final shape to flat pattern) to search for feasible bending sequences while the latter uses a forward reasoning approach (from flat pattern to final shape).

The most important task for bending process planning is the determination of bending sequence. The number of possible bending sequences is usually large even for a moderately complex part. Thus, enumerating and evaluating most or all possible sequences is not practical and sometimes not possible. As a result, the design is usually interpreted as a set of features during the planning stage. Some of the features suggest precedence constraints and heuristics for the feasible bending sequences. The precedence constraints and heuristics can then be used to reduce the search space and guide the search.

While the use of features looks inviting, because they provide useful encoding of known information, they can also cause new problems. For example, features can interact in negative ways. One feature may suggest process-a; another feature, process-b. This conflict raises questions of how these feature interactions can be detected and resolved. Hayes [8] used production rules to resolve negative interactions between features, and Nau et al. [12] used feature algebras to derive alternative interpretations of features to avoid them.

Our approach to deal with feature interaction problems, in sheet metal bending process, is twofold: first we observe and analyze the conflicts among the sub-systems of our planning system to detect the interactions, then we resolve the interactions by assigning precedences among the conflicting features.

Gupta and Nau [7] used machining features to analyze the manufacturability of machined parts. The evaluation results could be used as design feedbacks to speed up new product development. We discuss several manufacturing problems for bent sheet parts which are related to some special features. We show that by providing these feedbacks to the designers that we might be able to avoid these problems.

This paper is organized as follows, Section 2 gives the background of sheet metal design and planning. Section 3 briefly describes the design and bending process planning systems used in this paper. The features and their corresponding constraints and heuristics are then detailed in Section 4. Two example parts which are planned and manufactured by our system are given in Section 5.

In Section 6, the feature interaction and some manufacturing problems are discussed. Finally, The concluding remarks and our future plans are given.

2. BACKGROUND

2.1. Overview of sheet metal bending

Sheet metal bending operations involve placing sheet metal on a die up against a backgage to precisely locate the part. At this time, the machine is commanded to close the gap between the punch and die until the part is bent into the V-space of the die (see Fig. 1). In air-bending the part is not forced into the bottom of the V but rather is left in the air. This process causes less wear on the machine and tools than a bottoming (or coining) operation. When the part is taken out of the machine, the bend partially “springs back” by a small, but unknown amount which is usually determined by experiments.

The bending operation can either be done manually or automatically. We show, in Fig. 2, the automatic bending system (Amada BM100) used in this research, which has been augmented by our own open architecture controller. The system in our laboratory currently consists of a CNC press brake, a five-axis robot and a loader/unloader.

2.2. Conventions of sheet metal part drawing

The conventions for sheet metal part drawing are as follows: solid lines represent the boundaries of the part and the dotted lines or dashed lines, bend lines. In this paper, all numbers associated with the sheet metal parts in the figures are the indices of the corresponding bends instead of the actual bending sequences unless specified explicitly. The positive bend angles represent bent-up operations, and the negative ones, bent-down operations, with respect to the face normals of the flanges.

2.3. Current design and planning practice

State-of-the-art sheet metal parts design systems only represent the final part geometry. This geometry describes either the wire-frame or the boundary representation of a part. Unfolding software is then used to discover the topological relationships between the surfaces. With the unfolding result, the flat pattern with the bend deductions can be calculated [13]. The disadvantages of current design approach are the lack of feature information and the correlation between flat patterns and final part shapes can be ambiguous.

When the design is complete, manufacturing engineers manually identify features most useful for the process planning task. The plans are produced by human experts and often have to be adjusted several times in a trial-and-error process.

3. THE DESIGN AND PLANNING SYSTEMS

3.1. The design system

In our previous work [15], we developed the Parallel Design System called BendCad which is a design-with-features system that manages the relationships among the multiple representations of sheet metal parts. For example, when a flat pattern is bent, it is stretched in the process, which means that the part dimensions are effectively different in these two representations. Therefore, the system handles the correspondence between the models at the level of topology and cannot carelessly introduce or extinguish new faces or edges as the bending operation proceeds. One example part designed in BendCad is shown in Fig. 3.

Our system uses multiple representations of the sheet metal part to track the changes of the part during different manufacturing stages. A 1-D representation is used for the punching and nesting process, a 2-D representation is used for the punching and shearing processes, and a 3-D representation is used for the bending process. The final assembly of the sheet metal product is represented as the connectivity relationship between its 3-D parts, this relationship effectively adds a fourth dimension.

We have designed over 150 parts with BendCad to test our planning system. Other CAD designs can also be imported into BendCad as long as the data exchange files are provided. The geometric kernel of BendCad also serves as the geometric reasoning module for various sub-systems in our planning system.

3.2. The planning system

The process planning system developed at our laboratory consists of five primary sub-systems: an operations planner, a tooling system, a grasping system, a robot motion planner, and an open architecture controller. The architecture of the system is shown in Fig. 4.

Operations Planner: The planner generates possible bending sequences and asks the sub-systems to evaluate the manufacturing costs of these sequences. The costs are measured in terms of manufacturing time. An A* search algorithm is used to achieve near-minimum manufacturing costs. The A* algorithm [see Ref. 14 for details] is a heuristic search method that utilizes the heuristic estimate of the cost (usually denotes as the *h* cost) between the current state and the goal state. It is guaranteed to return a minimum-cost path if there exists one, and to return failure otherwise as long as the heuristic estimate is never over-estimated.

The operations planner is basically an A* search module. It generates partial bending sequences and asks various sub-systems for the *h* costs of the sequences. The initial bending sequence starts with the flat pattern (zero bend) and keeps expanding the most promising bending sequence until all bends are done. As the determination of bending sequence is a permutation problem, to make it more tractable, the precedence constraints and heuristics suggested by the features are used to prune the search space and guide the search. The precedence heuristics are used to adjust the costs of the nodes and the precedence constraints, to limit the breadth of the search tree. The root node of the search is the unbent part (flat pattern), and as the search proceeds, it generates the nodes which represent partial bending sequences (intermediate part shapes) until all bends are bent (final part shape) or the search fails.

Tooling System: The tooling system selects punches and dies, determines the number of tool stages (segments of punch-die pairs) and stage lengths, and performs interference checking between parts and tools. Finally, the system assigns each bend to a stage of tooling and lays out the complete set of tools on the machine.

Grasping System: The grasping system selects the grippers, determines the best grasping positions for the each bend, and predicts the number of repositions. These repositions are necessary whenever the gripper grasps across unbent bend lines or when there is no adequate clearance between the gripper and the bending machine. A reposition gripper holds the part and then the robot re-grasps the part to continue bending. Imagine a large four-bend box, we can sometimes bend three bends from just one grasping position, but we have to reposition the part and bend the last bend. As we plan for smaller and smaller parts, we will have to re-grasp the part twice and ultimately three times (one for each bend).

Motion Planner: The motion planner consists of two sub-systems: the “*Gross Motion Planner*” that determines the transfer motion of the robot, and the “*Fine Motion Planner*” that determines the motion of the robot when the part is inside the punch-die space, especially the retraction of the part after it is bent.

Open Architecture Controller: The bending machine’s controller has been replaced with an off-the-shelf engineering workstation that controls all of the low level machine operations. This high performance workstation makes it convenient to integrate all control activities with higher level planning activities, which have been traditionally kept separate. The controller manages all the real-time machine functions, interprets the plan (thus producing the part) and provides feedback so that the plan can be improved on an operation by operation basis. In some cases, this feedback process can nearly double the performance of the bending system.

Each sub-system is designed to cooperate with the operations planner in order to develop a near-optimal plan and the controller improves the plan further as parts are produced.

4. FEATURES AND PRECEDENCE RULES

4.1. Features of bent sheet metal part

Features are directly or indirectly related to the part geometry and can be used to various aspects of process planning. In this paper, we focus on the features for automatic bending process. These features either suggest precedence rules or the tool selection, grasping and motion strategies. We label sets of geometric entities as features and classify them as follows:

- *Bend graph*: This structure records the topological relationships between bends and the faces they connect. In bend graph, the faces are represented as nodes and the bend lines as links. Each link contains bending information such as bend angle, bend radius and bend deduction. The bend graph of a 2-bend part is shown in Fig. 5.
- *Internal tab*: An internal tab is a flange connected to a hole. For most sheet metal parts, it is desirable to bend the internal tabs first, or interference between the part and tools could occur in later bends. Internal tabs also pose problems when they are placed too near to other bend lines.
- *Essential and optional collinear bend*: Essential collinear bends (see Fig. 6(a)) are those bends with distinct bend lines that must be bent simultaneously to prevent distortion (e.g., a bend line over a hole). In most cases, optional collinear bends can be considered for efficiency (see Fig. 6(b)). In this example, we can perform the two collinear bends at once on a large tooling stage. By performing as many collinear bends as possible, the time savings can be considerable.
- *Outside/inside bend*: Outside and inside bends are labeled relative to the current grasping position (see Fig. 7), since the critical constraint is the angle of the bend relative to the grasping

plane. In our system, the plane of the gripper must match the plane of bend. Usually, outside bends should be bent first to avoid part interference with tools otherwise the part can “roll up” prohibiting tool access to the bend line. For instance, in Fig. 7(a), we apply the “outside bends first” rule and generate a feasible (interference-free) bending sequence. Once the bends are bent, the outside/inside relationship of the rest of the bends will be reconsidered. All outside bends have a depth information associated with them. The depth is defined as the number of faces between the grasping face and the outside bend. For example, in Fig. 7, bend 1 is labeled as outside bend with depth equal to 1. When bend 1 is bent, bend 2 becomes an outside bend with depth equal to 0. The outside bends with larger depths will have higher priority to be bent next than the ones with smaller depths if there are several outside bends to be considered.

- *Taller flange:* Some flanges can be labeled as tall flanges, if their heights are tall relative to other bends and the machine’s opening between the punch and die. The taller flanges are most likely to interfere with the machine if they are bent earlier, and so we prefer to postpone them as long as possible.
- *Shorter/longer bend*: Each bend is classified as either a shorter or longer bend according to the length of bend line and we try to bend shorter bends first. It is possible to bend shorter bends on longer tooling stages, but obviously we cannot bend longer bends on shorter tooling stages. The idea is to bend all short flanges on long tooling stages so that separate short tooling stages can be avoided. Consider a simple, rectangular, 4-bend box. If the long sides are bent first then tooling that “fits” must be used to bend the short sides, but when the “shorter bends first” heuristic is applied only the longer tooling is needed. This rule can significantly reduce the number of stages and the tool setup time, which is especially important when the part has many bends.
- *Channel:* A channel is a special feature for tool selection and bending sequence determination. The rule for recognizing channels is to look at the bend graph along a certain direction (for example, vertical or horizontal directions in the flat patterns) and match the patterns of bend directions. If all the bend angles are 90 degrees then the possible patterns for channels are (-, +, +) or (+, -, -). The channel suggests a particular ordering of the bending sequence that violates the “outside bend first” rule, so the ordering rule related to channels takes precedence. In Fig. 8, we show a channel feature and its feasible bending sequence (2 1 3). It is noted that the

"outside bends first" rule which suggests (3 2 1) will cause interference for the channel.

- *Corner*: Corners are common features for sheet metal parts. To compensate for the "spring back" characteristic of sheet metal, we usually need to overbend the bends. For parts with corners, the inside bends for the corner must be bent later otherwise the interference with outside bends of the corner will occur (see Fig. 9).
- *Hemming bend*: A hemming bend is a bend where the material is folded on top of itself or simply a bend with the bend angle of 180 or -180 degrees. A hemming bend also requires special tools (see Fig. 10(a)), and it is usually considered as a two-step (for instance, a 120 and 60 degrees combination) bend in the bend graph.
- *Large-radius bend*: Most V-die bends have bend radii smaller than twice the part thickness; however, larger-radius bends require special tools (see Fig. 10(b)).
- *Part overhang*: An overhang is a feature similar to a channel, but it is orthogonal to the current bend line. In this case, we choose special punches (hinged tool, see Fig. 11) that are designed to avoid the overhang in the unbent state and then to mechanically expand under the overhang during bending.
- *Louver and Dimple*: Pre-formed components for ventilation and assembly purposes. They are modeled as bounding boxes in our designs. Special care should be taken for louvers and dimples during the grasping position planning, tool interference checking and motion planning.

All the features described above will be generated automatically in our system. They are either labeled directly in BendCad or reasoned using the BendCad geometric kernel in the subsystems.

4.2. Precedence constraints and heuristics

In our system, features either suggest precedence rules or the tool selection, grasping and motion strategies. The precedence rules can be used as precedence heuristics or precedence constraints depending on the certainty factors of these relations. The certainty factors are determined by past planning expertise and geometric reasoning. For instance, the "outside bends first" rule is a strong heuristic compared with the "shorter bends first" rule. There are default precedence heuristics and constraints in the planning system and the system can change or overwrite these settings if necessary. Features also suggest special tools, workpiece grasping and fine motion strategies. These features are converted as constraints which are related to the bending sequences. The relationship between features, precedence heuristics and constraints, and other type of constraints is shown in Fig. 12.

Due to the combinatorial explosion characteristic of the search, for a part with N bends, the complexity of the ordering will be at least $N!*2^N$ (for each bend, there could be two possible holding positions). It is even worse if we take into account of repositions and collinear bends in the ordering. However, if we use the feature information and convert them into precedence constraints and heuristics, we can potentially avoid the combinatorial explosion and improve production efficiency

To utilize the precedence rules in the search for bending process plan, we assign different weights (actually, penalties that violate the rules) to these rules to indicate their relative strengths during the search. Unfortunately, none of the precedence rules is of 100% confidence in general. As a result, all the precedence rules are indeed precedence heuristics. To further reduce the search space, we have chosen few high confident precedence heuristics as precedence constraints. This usually speeds up the search dramatically. These precedence rules are usually set up prior to the search. However, the various sub-systems can issue the precedence rules during the search. The default setup of the precedence rules in our systems is shown in Table 1. It works pretty well among about 90% of the parts we tested. The rest 10% either have feature interaction or other manufacturing problems which will be explained in details in Section 6.

4.3. Precedence constraint language

We have developed a *precedence constraint language* to represent the precedence heuristics and constraints in our planning system. The language contains three syntactic forms:

numerical number: representing the bend index.

“”*: representing zero or more bend indices.

“?”: representing one and only one bend index.

For instance, if the constraint is “bend 1 is the first bend,” the corresponding representation is (1 *). If the constraint is “bend 1 precedes bend 5,” (* 1 * 5 *). Our system currently supports logical-and, logical-or and negation operators for the constraints. The heuristics are written the same way as the constraints except there is a penalty associated with each heuristic if the current sequences violate the heuristics.

4.4. Applications of the features in process planning

The various sorts of features used in our system can be used to eliminate expensive computation and reasoning, provide precedence heuristics and constraints for bending sequences, and suggest special tools. The goal of our system is to quickly generate the complete and near-optimal process plan that consists of the following components: bending sequence, tools selection and setup, workpiece grasping, and robot motion plan.

4.4.1 Bending Sequence Determination

Most of the features directly or indirectly relate to the bending sequence determination. We convert them into precedence heuristics or constraints to guide the search for a plan.

Several precedence rules are used to help select the bending sequences. For instance, the outside bends should precede inside bends, and internal tab bends and the shorter bends should be bent first. These rules suggest partial ordering of the complete bending sequence. The operations planner searches the sequences incrementally and obtains a near-minimum cost sequence if there is one.

To demonstrate how our planning system searches for the feasible bending sequence utilizing the precedence rules described previously, we show a simple 3-bend part in Fig. 13. The search starts with a flat pattern as its root node. The grasping position has been determined as shown to avoid repositioning. Bends 1 and 3 are labeled as outside bends and bend 2 is labeled as an inside bend. The system selects bend 1 to be bent first since bend 1 has larger depth than bend 3 (1 versus 0). As bend 1 is bent, bend 2 becomes an outside bend too. Now bends 2 and 3 are all outside bends with depth equal to 0. To reduce the amount of motion between consecutive bending operations, the neighboring bend of bend 1, i.e., bend 2 is chosen as the next bend. Finally bend 3 is selected and the complete bending sequence is generated. During the search, all sub-systems evaluate the states (partial bending sequences) to ensure the feasibility of the states. For this 3-bend part, the system generates a plan with the bending sequence (1 2 3), one tool stage and no reposition for the robot. Some states of the complete search space shown in Fig. 13 are not feasible and it is not necessary to expand them during the search.

4.4.2 Tools selection and setup

We select certain stages (pairs of punches and dies) to meet the bending requirements such as bending angle, bending radius, and material type. For each bend, we need to determine the number of stages and the stage lengths. Special tools should be selected for hemming bends and large radius bends (see Fig. 10). We have to treat these features as constraints in selecting the tools during the planning in order to satisfy the part design.

For the given bending sequence, the initial shape begins with flat pattern, and the intermediate shape is simulated before and after the bend in the punch-die space. We then check the collisions between the part and tools.

Dimples and louvers are represented as bounding boxes in the designs, so conservative estimates can be made for detecting collisions between the part and tools, or the part itself.

4.4.3 Workpiece grasping

In our bending system, a five-axis robot is holding and positioning the part. In order to determine better grasping positions for each bend, the grasping system must choose appropriate grippers for parts of different geometry. Usually, holes and cutouts are considered as bad grasping areas if the contact area between the gripper and the part is less than about 80% of the gripper area. The grippers have limited knuckle heights, so they can only grasp across short bent flanges. Grasping dimples or louvers must also be avoided. Special steps are taken dealing with hemming bends and large-radius bends, once they are bent.

4.4.4 Robot motion plan

For a given bending sequence, the intermediate shape of the part is calculated, and the bounding box approximation of the intermediate shape is used for planning the gross motion of the robot. Since we have a sparse environment, we can use the bounding box approximation for the robot gross motion planning.

When the workpiece is going into or retracting the punch and die space, this step of the process is referred to as fine motion planning. To plan the path more efficiently, we must identify the positions and orientations of dimples, louvers and bent internal tabs. The fine motion planning usually couples with tools selection in our system.

5. PLANNING EXAMPLES

In this Section, two example parts are used to produce the complete bending process plans. The precedence constraints and heuristics are set up as in Table 1. The first example (Fig. 14) is a 7-bend part which takes about 10 minutes to plan. While the second one (Fig. 15) is a complicated 23-bend part which takes about 100 minutes to plan. After these plans are generated, they are used to make the parts on our bending machine. Appendix A shows the partial listing of the plan for the first example.

6. FEATURE INTERACTION AND SOME MANUFACTURING PROBLEMS

6.1. Feature interactions

The major problem of using features in process planning is the interaction of features, especially negative interactions, i.e., bending one feature first makes it difficult or impossible to bend subsequent ones. The main reason for this problem is that most planning systems deal with features independently. While in some cases, several features have to be taken into account altogether in order to produce feasible process plans. These interactions could either make a part difficult (i.e., taking longer time) to plan or sometimes impossible to plan. In our system, the interactions could occur between the precedence rules used in the search or among the sub-systems while each of them trying to optimize their own goals. Interactions can be detected when the sub-systems begin to return costs of infinity, or the search backtracks too often. Once we identify feature interactions, we will run each sub-system separately with the operations planner. Analyze and compare with results from each sub-system and then identify the conflicting features. We then resolve the interactions by assigning precedence among the partial sequences suggested by the conflicting features. Currently, this is done manually in our system.

After having tested many parts, we have verified that the interactions of features are highly geometry-dependent. For instance, part 1 in Fig. 16(a), bends 2 and 3 are tall bends with respect to bends 9 and 10. The “tall bends later” rule applies for this part, i.e., (((* 9 * 2 *) (* 10 * 2 *) (* 9 * 3 *) (* 10 * 3 *))). Now, let us look at part 2 (Fig. 16(b)) which has similar geometry as the previous part. However, if we use the same precedences, we cannot bends 2 or 3 if bends 9 or 10 are already bent, since we cannot place the part on the die (the flanges f1, f2, f3 and f4 will interfere with the tools). In this case, we should bend 2 and 3 before bends 9 and 10 to resolve the interaction, i.e., (((* 2 * 9 *) (* 3 * 9 *) (* 2 * 10 *) (* 3 * 10 *))).

Another example of feature interaction is shown in Fig. 17(a) and 17(b), these two parts are almost identical except the positions of “Tab1” are different (for part 4, “Tab 1” is very close to bend 3). Each part has two internal tab bends (bends 4 and 5). According to the “internal tab bends first” rule, bends 4 and 5 should be bent before bend 3. This rule applies for part 3 in Fig. 17(a). However, for part 4 in Fig. 17(b), if we bend 4 first, we can never bend 3 (“Tab1” will interfere with the tools). Instead, we should bend 5 first and bend 3 before 4, i.e., (* 5 * 3 * 4 *).

We have set up the planning environment for these parts discussed above. The partial precedence settings are listed in Table 2. In Table 3, we show the search results on a SunSparc 10 workstation. We compare the *number of nodes* visited in the A* search space. Each node represents a partial bending sequence in the search. In these results, we run the operations planner, grasping system and tooling system, since these three sub-systems use these features most extensively. The results show that parts with almost identical shapes can result in totally different bending sequences due to the feature interactions. For parts 1 and 3, our system can find the bending sequences efficiently since there is no feature interaction. But, for parts 2 and 4, the search backtracks a lot due to the feature interactions. We can tell the difference by comparing the search results in Table 3 (parts 1 and 2, 79 versus 44 nodes, and parts 3 and 4, 191 versus 20 nodes). As we discussed above, bend 3 in part 4 should be bent before bend 4, i.e., ((*3 * 4 *)). We further assign this constraint in the search and save about 90% (191 versus 21 nodes) of nodes visited. This shows the power of using constraints to resolve the interactions. However, there is a trade-off of using precedence constraints in the search. It could help the search dramatically for interaction-free parts. However, it could also fail the search for parts with interactions. Since the search for feasible bending sequences is basically a combinatorial problem, we need to have some precedence constraints to make the search more tractable (i.e., to find a feasible sequence in a reasonable amount of time). There are few rules we are confident enough to use them as the default precedence constraints. To develop and evaluate these constraints, we do a case-by-case study of process geometry and limitations, and convert them into precedence constraints either before planning or during the planning. The rest of precedence rules are treated as heuristics in the search.

We found that even if we convert features into constraints, sometimes, we still cannot avoid combinatorial explosion. In Fig. 18, there are interactions between two channels and two possible collinear bends. There are combinations of bending all channels at a time, or one at a time. The possible collinear bends have similar combinations. The problems occur when we try to group the same kind of features as a single feature, for instance, try to bend two channels at a time as a single channel and three collinear bends as one collinear bend.

Grouping features might be favorable because of its efficiency but it will make planning more difficult since we have to consider the all of the subsets of constituent features, which will cause a new combinatorial problem.

6.2. Some manufacturing problems

Based on the geometric features and the physical limits of the manufacturing system we have thus far described, we now consider the problem of offering suggestions to the designer in order to resolve the most pressing manufacturing problems we encountered.

The 8-bend box (see Fig. 19) illustrates an example of the “overhang” feature that requires special “hinged tool.” This tooling has the property of expanding under part overhangs, once the tooling touches the material, and then contracting when the tool is removed from the bend (see Fig. 11). Unfortunately, this tooling is expensive and often not available.

Before searching for an adequate bending sequence, the features are identified and it is noted that “hinged tooling” may be required. When the search arrives at the final bends then the models for this complex tooling is applied, but only after the simple tooling has failed. In the case of the 8-bend box, this tooling must be used since there are overhangs in last bends of all possible bending sequences. However, there are parts, such as a 6-bend box (for instance, taking out the two flanges f1 and f2 of the 8-bend box in Fig. 19), where the hinged tooling is not required and one bending sequence may suggest special tooling, while another sequence would not. All else being equal, the tooling system avoids special tooling, but often the grasping or motion systems may find constraints where it continues to be necessary for reasons “external” to tooling.

Let us consider the problems of grasping the 6-bend box (similar to the 6-bend box just considered). If the part is large enough then the grasping system can do all of the bends with only one reposition by front loading the part into the bending machine and then bending the other two bends by side loading on either side of the current grasping position.

In these cases, the search for an adequate bending sequence solution can become protracted and rather than trying to solve a complex multi-objective function in an exponential search space, it may be more appropriate to present this dilemma to the designer and ask him/her to make modifications that resolve the dilemma. However, the suggestion is complex and must take on a form that shows the chain of problems:

“part-dimension too small -> 2 repositions” and *“2 repositions -> special hinged tooling”*

By providing the problem chain, the designer can quickly understand the “root” cause and preempt the problems at the most appropriate level. Note that the designer may choose to intervene at any level.

Gripper Selection - Often part’s design has subtle features, which cause problems picking an adequate gripper for material handling. This is caused by the flange heights and the fact that the knuckle must be high enough to span a particular flange. This problem is further aggravated by geometry of the gripper and how close it can be to the machine during the bending operation without collision. For example, on our machine, a standard gripper can get as close as 120 mm from the machine when the gripper is right-side-up and when the gripper is upside-down it can be no closer than 240 mm. This means that if there are down flanges (with respect to the current bend angle) then the gripper is extremely limited to how close it can get to the machine. This problem often leads us to want to choose a gripper with knuckles on the opposite jaw of the gripper (pointed down) from the majority of bends, so that it would not collide with the machine when it is doing those bends.

This problem arises in production of a relatively simple part (see Fig. 20), and it is difficult to identify, because the designs offer little clue that a special gripper is required. But as the search for the solution proceeds, it becomes clear that all of the “up-knuckle” grippers have no chance to make the part in a reasonable plan. The problem is caused by the large difference in the minimal approach values for the two orientations of the gripper, and this feature only yields to considerable dimensional checking. The last straw is that this resource may or may not be available.

Unlike the previous example, it is unlikely that the designer would be willing to change all of the bend angles to suit the manufacturing process. However, it may be appropriate to suggest that a “down-knuckle gripper” be purchased for the purpose of making a part.

Tool Availability - Sometimes when a part is produced, the flanges penetrate the tool space in every bending sequence. But with small changes in the flange heights it is possible to avoid this problem. The other more costly solution is to design custom tooling, where tools are trimmed to avoid collisions. This solution is often desirable when there are form features (louvres or dimples) in high value parts. If these features come into contact with the machine during the bending process, it may be desirable to allow extra clearance for them.

Die Rail Length Limit - When the punches and dies are laid out on the bending machine, there must be clearance gaps between them and the tooling lengths must be assigned to an appropriate bend. This layout task must all be computed to minimize the overall length of tooling on the machine.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we identify several important features for bent sheet metal parts and use them in the production of process plans. These features suggest precedence rules, or constraints for tool selection, workpiece grasping and motion strategies. We then convert these features into corresponding heuristics and constraints to make the search of bending plan more tractable.

Occasionally, features interact in a way that makes both the planning and manufacturing process more difficult than might be necessary. This is still known as an open problem of using features in process planning. We show two examples and resolve the interactions by assigning precedences among these conflicting features. This is done manually by using past planning expertise and studying the results from the individual sub-systems. We also show several manufacturing problems that might be used as design feedbacks in order to avoid them.

We have applied our planning system on many complicated parts, and the results are satisfactory. Our approach has been applied successfully on the parts that used to be considered as difficult or impossible to plan for human experts. Currently, we are able to plan and manufacture the majority of over 150 parts we have tested under one hour after the flat patterns are prepared.

Our future work will focus on resolving feature interaction problems automatically by sharing constraints among various sub-systems and making use of design feedbacks from the manufacturing perspectives to simplify the planning.

ACKNOWLEDGMENT

This research is partially supported by Amada Inc. We would like to thank Amada for providing current practice and experience on sheet-metal parts design, manufacturing and planning. We thank Mr. Ken Hazama for his constructive suggestions. We also want to acknowledge the contributions by our project members: S. K. Gupta, Kyoung Kim, S. S. Krishnan, Duane Williams, and Richard Moore who have developed the various sub-systems with the authors.

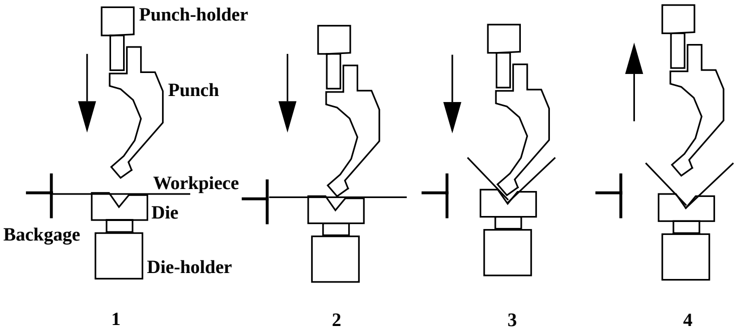


Fig. 1. Bending operation.

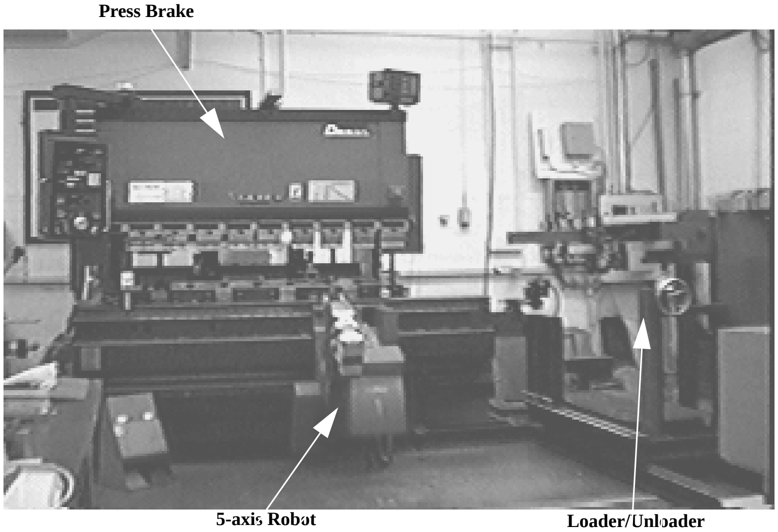
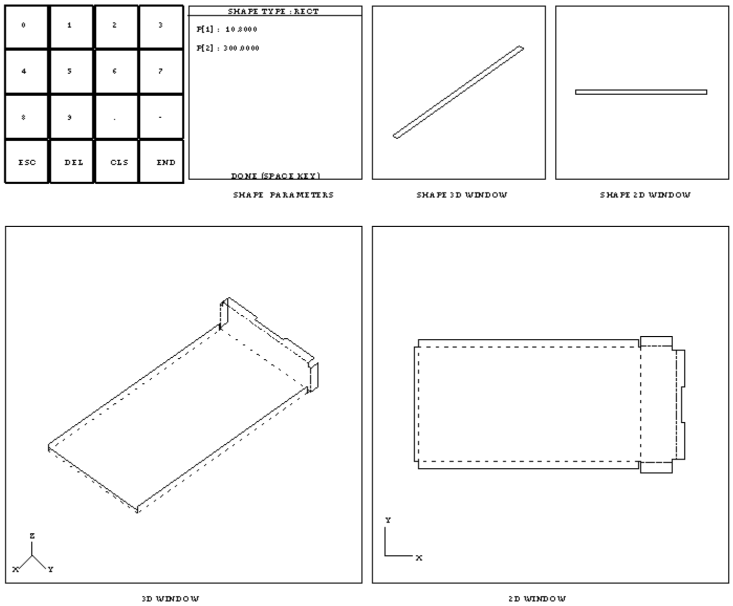


Fig. 2. Amada BM100 bending system.



SHAPE TYPE : RECT

P[1] : 10.0000

P[2] : 300.0000

Fig. 3. An example part designed using BendCad.

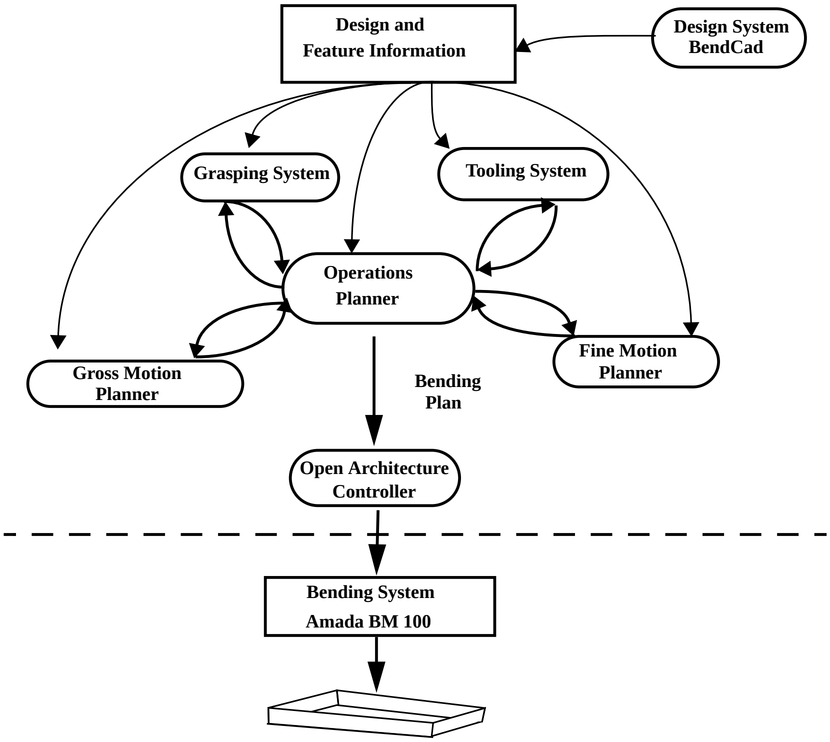


Fig. 4. Architecture of the automatic bending process planner.

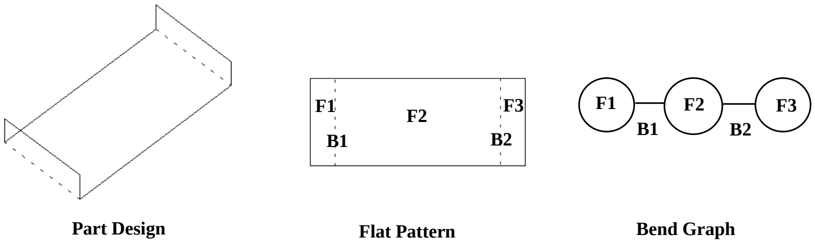


Fig. 5. Bend graph of a two-bend part.

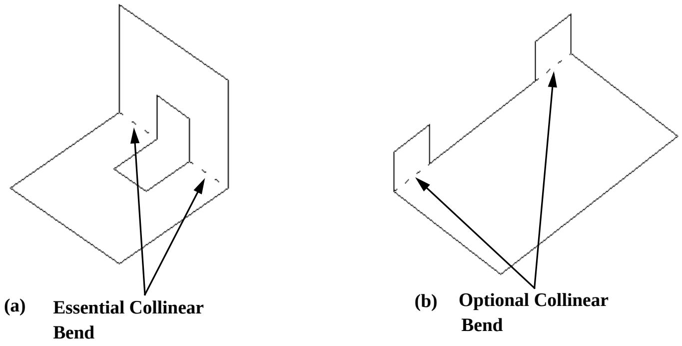


Fig. 6. (a) Essential collinear bend and (b) Optional collinear bend.

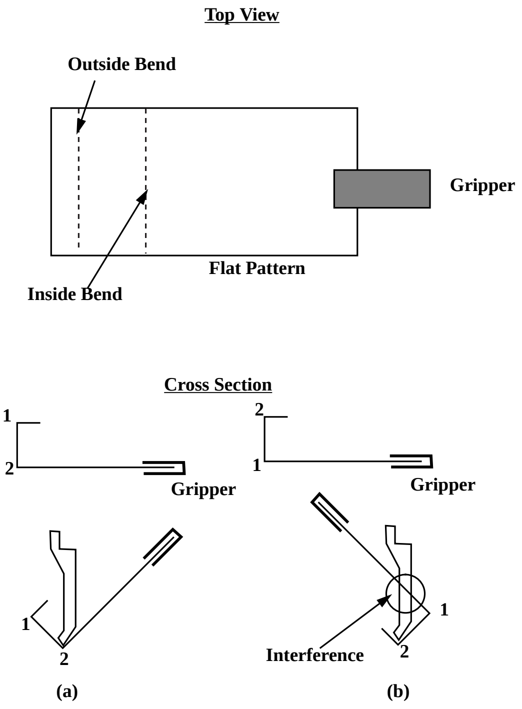


Fig. 7. (a). Feasible bending sequence applying the outside-bends-first rule. (b). Infeasible bending sequence which violates the outside-bends-first rule. (numbers shown here are bending sequences).

Fig. 8. Cross section of a channel feature with the feasible bending sequence (2 1 3).

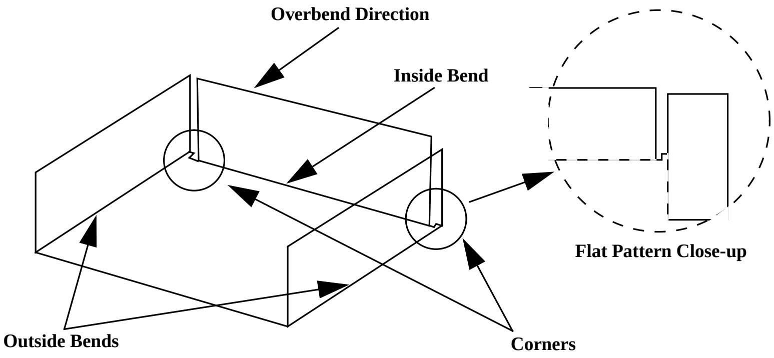


Fig. 9. Corners.

(a). Hemming Bend

(b). Large-radius Bend

Fig. 10. Special tools.

Fig. 11. Hinged tool.

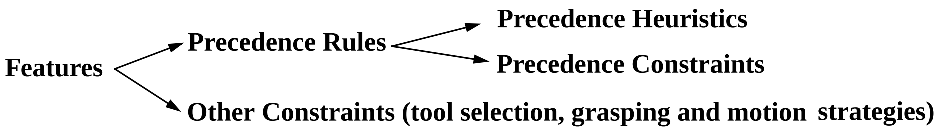


Fig. 12. Relationships between features, heuristic and constraints.

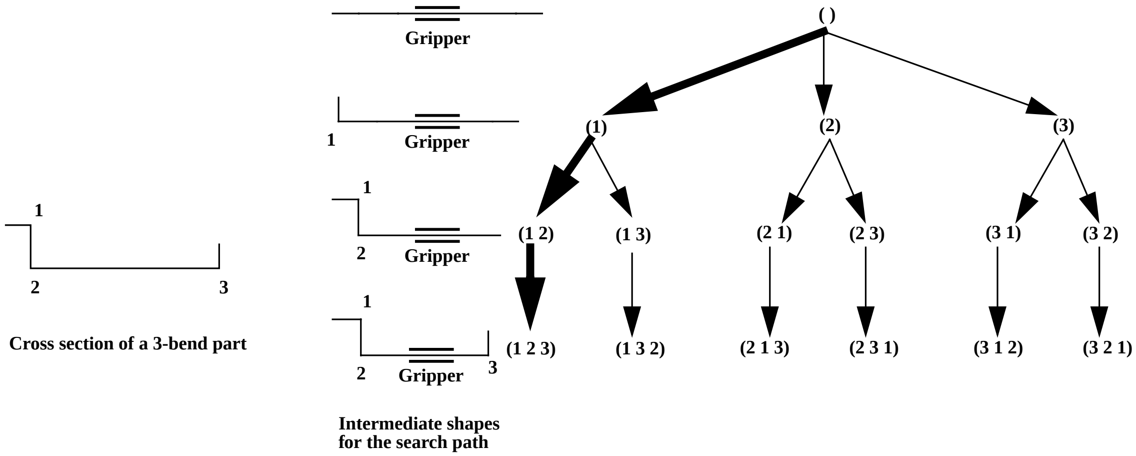


Fig. 13. The complete search space of a 3-bend part.(the thick line indicates the search path for the bending sequence)

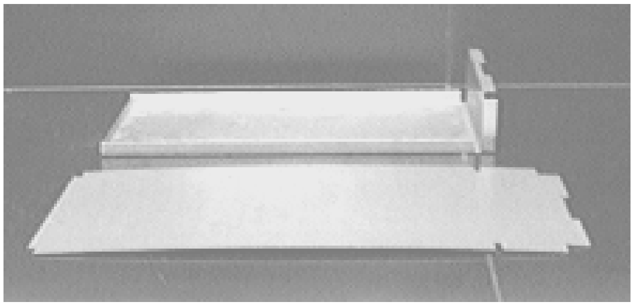


Fig. 14. The flat pattern and final part shape of the first example part.

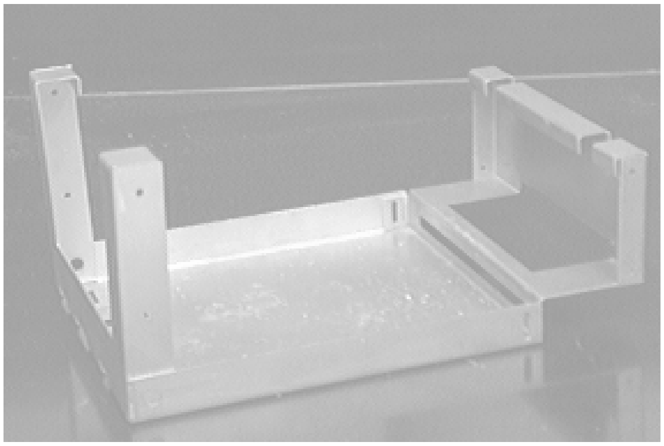
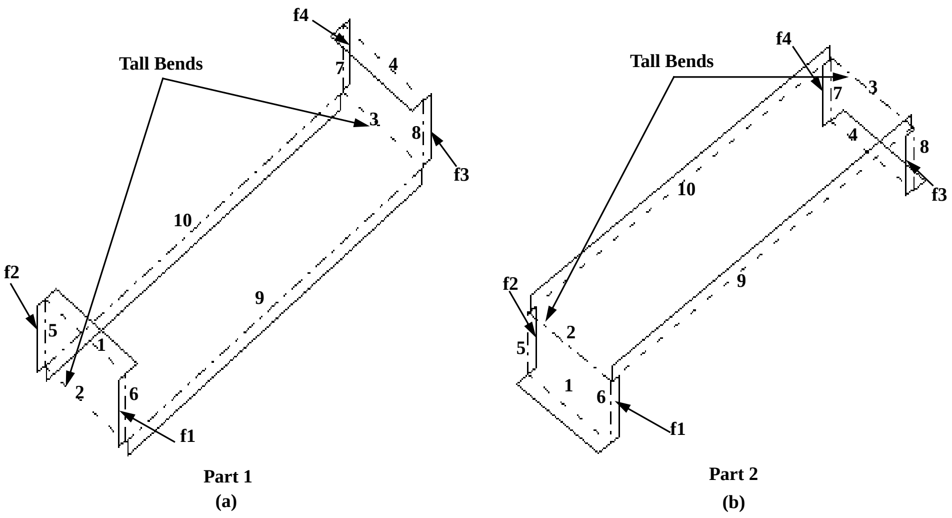


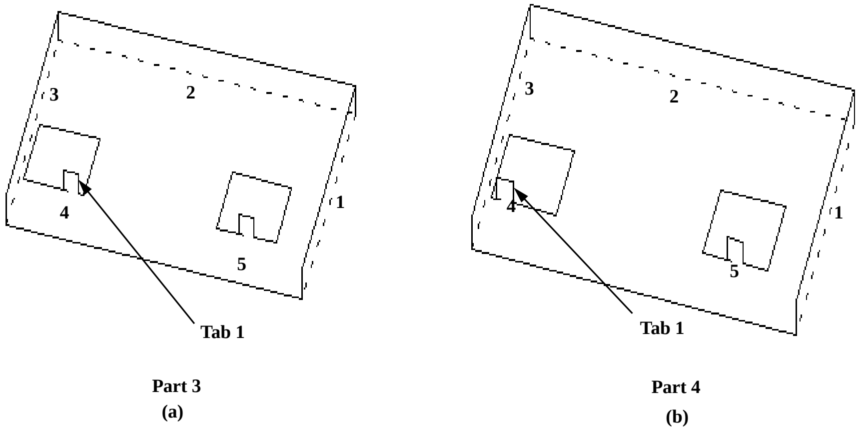
Fig. 15. The second example part: a complex 23-bend part.



Part 1
(a)

Part 2
(b)

Fig. 16. (a) and (b): Feature interaction of tall bends.



Part 3
(a)

Part 4
(b)

Fig. 17. (a) and (b): Feature interaction of internal tab bends.

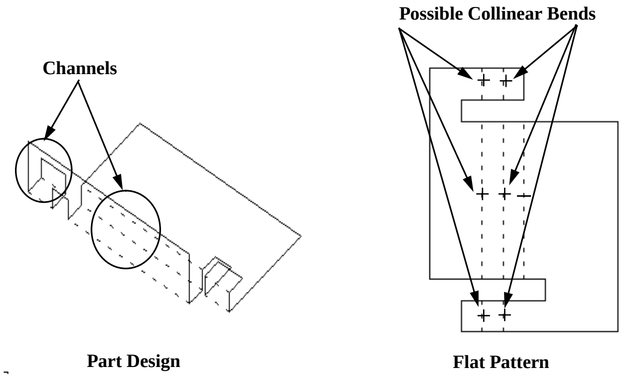


Fig. 18. Combinatorial problems for grouping features (the “+” sign represents bent-up operation and the “-” sign, bent-down operation).

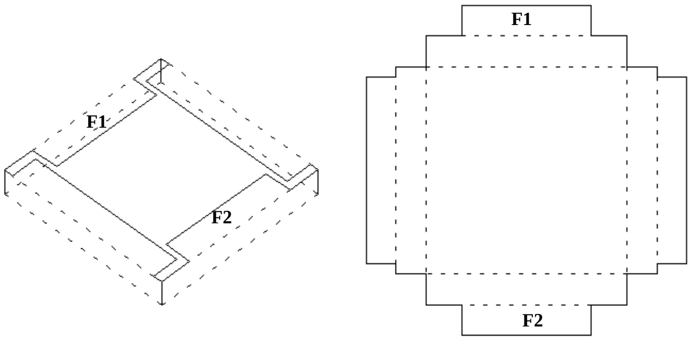


Fig. 19. Part shape and flat pattern of an 8-bend box which requires hinged tool. (see Fig. 11)

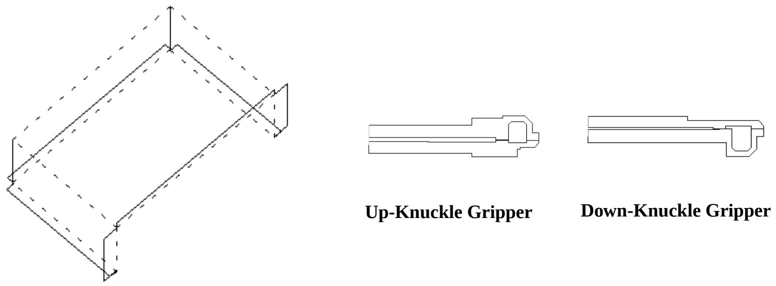


Fig. 20. Part shape of a down-knuckle-gripper part.

Table 1. Default precedence rules and heuristic penalties.

Precedence Constraints	outside bends first	inside bend first for corners	
Precedence Heuristics	tall bends later (penalty: 6)	internal tab bends first (penalty: 6)	long bends first (penalty: 3)

Table 2. Partial settings for the planning examples.

Part Number	Part 1 (Fig. 16(a))	Part 2 (Fig. 16(b))	Part 3 (Fig. 17(a))	Part 4 (Fig. 17(b))
No. of Bends	10	10	5	5
Default Precedence Heuristics	((* 9 * 2 *) 6) ((* 9 * 3 *) 6)	((* 10 * 2 *) 6) ((* 10 * 3 *) 6)	((* 4 * 3 *) 6)	((* 5 * 3 *) 6)
Assigned Precedence Constraints				((* 3 * 4 *))

Table 3. Search results.

Search Space	Part 1 (Fig. 16(a))	Part 2 (Fig. 16(b))	Part 3 (Fig. 17a))	Part 4 (Fig. 17(b))
Planned Bending Sequence	(8 7 6 5 4 1 10 9 3 2)	(8 7 6 5 4 3 1 2 10 9)	(5 4 3 2 1)	(5 3 4 2 1)
Using *Default Settings*	44	79	20	191
Using *Assigned Constraint*				21

Appendix A: Partial bending plan (in an attribute-value format) for the part in Fig. 14

```
(REPORT ((TYPE MESSAGE) (FROM PLANNING) (TO INTERACTING) (STATE REPLY))
((TYPE FINALIZE)

; OPERATIONS PLANNING MODULE
; Final Bending Sequence; Elements in the sequence: (Bend_Index Grasping_Face_No), (0 0) represents a Reposition, (-10 0) represents end of the bending sequence

 (BENDS ((5 5) (6 5) (1 1) (0 0) (7 5) (3 1) (4 1) (2 1) (-10 0)))


; GRASPING MODULE
; Grippers Selection
 (ROBOT_GRIPPER “IG-0725-U332Q”)
 (REPO_GRIPPER “YG-1521-U59N”)
...
; Loading/Unloading Positions
  (LOADER_LOCATIONS ((48.00 48.00) (328.00 138.00)))
 (UNLOAD_LOCATIONS ((48.00 77.80) (283.00 77.80)))

; Robot Grasping Positions
 (ROBOT_LOC ((1 -1 (300.00 166.40 0.00) (0.00 166.40 0.00) 190.00 68.60 137.39 0.00 1 0 0 67.01 2.00) (1 1 (0.00 0.00 0.00) (300.00 0.00 0.00) 160.00 21.00 26.00 11.80 0 0 0 56.17 4.00)))
; Robot Reposition Locations
 (REPO_LOC ((1 1 (300.00 166.40 0.00) (0.00 166.40 0.00) 53.00 52.30 57.60 0.00 74.76 3.14 2.00)))
...

; TOOLING MODULE
; Punch (Punch Holder) /Die (Die Holder) Selection and Stage Layout
((TYPE PUNCH_SEGMENTS)
 (STAGE 1)
 (LENGTH 150)
 (HORNS “NONE”)
 (SIZES (100 40 10))
 (X_LOCATION 525.00)
 (PUNCH “00301”)
 (PUNCH_ORIENT 2)
 (PUNCH_HOLDER “00001”))
...

((TYPE SETUP)
  (BACKGAGE_FINGER_LENGTH 75.00)
 (PHL_LOCATIONS (75 275 475 675 875 1075 1275 1475 1675 1875 2075 2275))
 (PHL_TYPES (150 150 150 150 150 150 150 150 150 150 150 150))
 (DHL_LOCATIONS (0))
 (DHL_TYPES (2500))
 (STAGE_LOCATIONS (525 1150 1525))
 (PUNCH_TYPES (“00301” “00301” “00301”))
 (DIE_TYPES (“10608” “10608” “10608”))
```

```
; Stage Layout
 (STAGES ((2 150.00 525.00 2 0 “00301” “10608” “00001” “53150” 2 1 1 5.00 5.00 0 1 0) (1 300.00 1150.00 2 0 “00301” “10608” “00001” “53150” 1 1 1 311.60 50.60 2 3 0) (3 45.00 1525.00 2 0 “00301” “10608” “00001” “53150” 3 1 1 5.40 308.00 1 0 0)))))

;MOTION PLANNING MODULE
; Robot Fly Points
((TYPE ROBOT)
 (GRIPPER CLOSE)
 (ROBOT_SPEED 3)
 (VERIFY_POINT (1017.00 -426.57 124.56 -18.99 85.40))
 (ROBOT_ABS_MOVE ((500.00 -400.00 100.00 0.00 -90.00))))
((TYPE BACKGAGE)))
(GET ((TYPE MESSAGE) (FROM PLANNING) (TO SEQUENCING) (STATE REQUEST))
((TYPE LOADER)
 (ROBOT_ABS_MOVE ((900.00 -500.00 105.00 0.00 -90.00) (900.00 -500.00 105.00 -90.00 -90.00) (900.00 -500.00 105.00 -180.00 -90.00) (900.00 -598.80 105.00 -180.00 -90.00) (889.17 -598.80 499.20 -180.00 -90.00)))
 (ROBOT_ZGAGE (518.20 480.20))
 (MEMORIZE_PRELOAD_POINT (889.17 -598.80 499.20 -180.00 -90.00))
 (ROBOT_REL_MOVE ((268.60 0.00 0.00 0.00 0.00)))
 (ROBOT_ABS_MOVE ((1157.77 -598.80 480.20 -180.00 -90.00)))
 (EXCHANGE_PART 5.00)
 (ROBOT_REL_MOVE ((0.00 0.00 6.50 0.00 0.00)))
 (ROBOT_REL_MOVE ((-300.00 0.00 0.00 0.00 0.00)))))
...
(BEND ((TYPE MESSAGE) (FROM PLANNING) (TO SEQUENCING) (STATE REQUEST))
((TYPE FOLLOW)
 (BEND (4 1))
 (BEND_ANGLE 90.00)
 (MATERIAL_THICKNESS 1.00)
 (PRESS_TRAVEL 2.57)
 (FOLLOWING_SPEED 0)
 (FOLLOWING_HEIGHT 0.00)
 (MODE DIE_CONTACT)
 (BGAGE_ABS_MOVE ((188.00 110.00 -92.00 110.00 33.38)))))
...
```

REFERENCES

1. Anderson, D.C., Chang, T.C.: Geometric Reasoning in Feature-based Design and Process Planning. *Computer & Graphics*, Vol.14, No. 2, pp. 225-235, 1990.

2. Bourne, D.A.: Intelligent Manufacturing Workstations. In *Knowledge-Based Automation of Processes, ASME Winter Annual Meeting*, Anaheim, CA, pp. 77-84, 1992.

3. Cutkosky, M.R., Brown, D.R., Tenenbaum, J.M.: Extending Concurrent Product and Process Design Toward Earlier Design Stages. *Concurrent Product and Process Design*, Chao and Lu (eds.). ASME DE-Vol. 21, PED-Vol 36, 1989, pp. 65-72.

4. de Vin, L.J. et. al: PART-S, a CAPP System for Small Batch Manufacturing of Sheet Metal Components. In *Proceedings of the 24th CIRP International Seminar on Manufacturing Systems*, Copenhagen, pp. 171-182, 1992

5. Dixon, J.R.: Designing with Features: Building Manufacturing Knowledge into More Intelligent CAD Systems. In *Proceedings of ASME Manufacturing international-88*, Atlanta, GA, April.

6. Floriani, L.De: Feature Extraction from Boundary Models of Three-Dimensional Objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 8, August, 1989.

7. Gupta, S.K., Nau, D.S.: Systematic Approach to Analyzing the Manufacturability of Machined Parts. *Computer-Aided Design*, Vol. 27, No. 5, pp. 323-342, 1995.

8. Hayes, C.: Using Goal Interactions to Guide Planning. In *Proceedings of AAAI-87; the Sixth National Conference on Artificial Intelligence*, pp. 224-228, 1987.

9. Inui, M., Kimura, F.: Design of Machining Processes with Dynamics Manipulation of Product Models. In *Artificial Intelligence in Design,* Pham, D.T. (ed.). Springer-Verlag, New York, pp. 195-227, 1991

10. Kurochi, N. et al.: CAD/CAM System for Sheet Metal Structural Parts: Development and Implementation. *Bull. Japan Soc. of Prec. Eng.*, Vol. 13, No. 3, Sep. 1979.

11. Nau, D.S. et al.: Solid Modeling and Geometric Reasoning for Design and Process Planning. *Tech. Report CS-TR-2056,* University of Maryland, July, 1988.

12. Nau, D.S., Karinthi, R.R.: An Algebraic Approach to Feature Interactions. *Technical Report ERC-UMC 89-101*, University of Maryland.

13. Reich, R., Ochs, J.B., Ozsoy, T.M.: Automated Flat Pattern Layout from Three-dimensional Wire-Frame Data. *Journal of Engineering Design*, Vol. 2, No. 3, 1991.

14. Rich, Elaine and Knight, Kevin: Artificial Intelligence, McGraw-Hill, New York, 1991.

15. Wang, C.-H., Sturges, R.H.: BendCad: a design system for current multiple representations of parts, *Journal of Intelligent Manufacturing*, Vol. 7, pp.133-144, 1996

16. Zussman E., Horsch T.: A Planning Approach for Robot-Assisted Multiple-Bent Profile Handling. *Robotics and Computer-Integrated Manufacturing*, Vol. 11, No. 1, pp. 35-40, 1994.