

DETC98/CIE-5708

SKETCH-N-MAKE: AUTOMATED MACHINING OF CAD SKETCHES

Mark Bloomenthal ¹ University of Utah mark-b@cs.utah.edu	Robert Zeleznik ² Brown University bcz@cs.brown.edu	Russell Fish ¹ University of Utah fish@cs.utah.edu	Loring Holden ² Brown University lsh@cs.brown.edu	Andrew Forsberg ² Brown University asf@cs.brown.edu	Richard Riesenfeld ¹ University of Utah rfr@cs.utah.edu
	Matt Cutts ³ UNC Chapel Hill cutts@cs.unc.edu	Samuel Drake ¹ University of Utah drake@cs.utah.edu	Henry Fuchs ³ UNC Chapel Hill fuchs@cs.unc.edu	Elaine Cohen ¹ University of Utah cohen@cs.utah.edu	

ABSTRACT

Sketch-N-Make is a prototype system that enables individuals to create dimensioned designs quickly for a restricted, but broadly useful, domain of machined metal and plastic prismatic parts. Process plans, fixturing, tool selection, feeds and speeds, and toolpaths are automatically generated from the designs allowing parts to be manufactured on a machining center with the aid only of a technician who prepares and loads the stock and cutting tools.

The system employs a gestural interface that exploits familiar pencil sketching paradigms to allow users to create and manipulate geometry without specifying the details required for manufacturing. The system transforms the geometry into a high-level feature-based model that represents the geometry and additional information from which machining features are derived.

The sketching interface has mechanisms to constrain and parameterize 3D model geometry in ways required for precise part design and design iteration. The user can modify iteratively the geometric representation causing the system to automatically update the feature-based model accordingly. Then the system generates the necessary machining operations to manufacture the geometry on a machining center according to a family of process plans.

The synergy between the choices for part domain, user interface, and process plan family helps provide a smooth and rapid

pathway from concept to detailed design to machined metal or plastic parts.

INTRODUCTION

Sketch-N-Make is a prototype *art-to-part* system that allows users to quickly design a non-trivial class of dimensioned machined metal and plastic prismatic parts. These parts may then be automatically manufactured on a machining center with the aid only of a technician who prepares and loads the stock and cutting tools.

One goal for this system is to allow a broader class of users, those without manufacturing expertise, to produce manufacturable designs more rapidly and with less expense. Another goal is to provide a smooth pathway from original concept through finished machined part. Along this pathway, we want to allow the user to concentrate primarily on the geometric and parametric aspects of the design and be concerned only minimally with issues of its manufacture. The process plan, fixturing, tool selection, feeds and speeds, and toolpaths necessary to actually manufacture the part are then generated automatically from the design.

One set of impediments to attaining these goals follows from the way in which users typically interact with CAD systems. Many designers sketch with pencil and paper as the preferred, unencumbered, way to visualize early stage geometric concepts. However, most CAD systems require detailed and complete specifications for designs. Such sys-

¹Department of Computer Science, University of Utah, 50 South Central Campus Drive, Room 3190, Salt Lake City, Utah 84112-9205

²Brown University, Box 1910 Department of Computer Science, Providence, RI 02912

³Department of Computer Science, University of North Carolina at Chapel Hill, Sitterson Hall, CB 3175, Chapel Hill, NC 27599

tems typically have user interfaces that are oriented towards WIMP (Windows, Icons, Menus, and Point-and-click) interaction which make little or no use of users' often well developed abilities to provide information by gesturing or sketching. Further, once design details are known and entered as part of the design, changes to the design will need to take these details into account thereafter. Attempts to subsequently manipulate designs on an abstract level may become obscured by a clutter of details.

Other obstacles to achieving these goals follow from problems of automatically manufacturing from designs. These problems include:

Interpretation: The geometry of the design must be interpreted into a set of machining operations that can produce the geometry to required tolerances using a subset of available manufacturing processes.

Process Plan: A strategy for selecting and sequencing these machining operations must be determined. This strategy, or *process plan*, breaks the manufacturing into a sequence of stages. Each stage consists of an ordered set of machining operations to be carried out on a particular machine tool using a particular set of cutting tools. Each stage also specifies the fixturing with which the part will be held in a given orientation.

Fixturing: Integral to the process plan is a strategy for how to hold the part, or workpiece, securely as it is being machined in each stage. This strategy may require temporary fixturing components and modifications to the workpiece itself. The fixturing must not interfere with the machining operations in the process plan or the eventual function of the part.

Accessibility: The cutting tool must have interference-free access to the portion of the workpiece to be cut.

Finding **any** solutions to these manufacturing problems for arbitrary geometry may be difficult or impossible, and automating the process of finding solutions is even more difficult. The situation is complicated by interdependencies between these problems and hence techniques to solve them are usually iterative. Even for fairly simple geometry, solutions are typically not unique and may be difficult to find algorithmically. Optimization of time and cost may also be critical factors in solving these problems.

Our prototype system uses a carefully matched set of elements applied to the design of a non-trivial class of prismatic parts currently restricted to a single machining access direction. Part designs of this nature have wide practical use. This part domain allows for the selection of user interface and model transformation techniques that, in combination with a family of process plans, result in tractable solutions to the obstacles presented above.

RELATED WORK

Since this paper reports research on an art-to-part system, related research spans topics from user interfaces through manufacturability analysis. We have found no single paper reporting a completely comprehensive approach to rapid functional part realization spanning this range of topics, although there are many related research results on various combinations of subsets of problems. Reviewing all such work is beyond the scope of this paper. We attempt here to indicate some highlights.

There has been recent work which has focused on using gestures to specify 3D geometric models in contrast to typical WIMP interaction with CAD systems. These "sketching" systems have demonstrated the potential for compressing long sequences of menu based CAD constructions into rapid and fluid gestures.

Sketch (Zelevnik,1996) was one of the first systems to use gestures as the sole interface to a 3D modeler. The system uses a natural "pencil and paper" paradigm where gestures are embedded in the 3D scene. The Sketch system demonstrated that all of the basic functions of a 3D modeling system could be accomplished without a WIMP interface. However, Sketch was aimed primarily at architectural design issues.

Others have used sketches to infer 3D models. Quicksketch (Eggl,1995) allows gestural creation of objects with inferred constraints which the system attempts to maintain during gestural editing. The 2D shape inferred is based on the shape that is drawn, rather than a vocabulary of gestures, while the extension to 3D has some similar capabilities to the Sketch approach. This system allows under-constrained sketches to be disambiguated either by a set of heuristics or by additional input from the user. The system was targeted mainly towards preliminary mechanical CAD models but did not do precise dimensional modeling. In (Grimstead,1995) 3D solid B-rep models were created from a single hidden line removed view of a drawn 3D object. Topology is inferred and a least square solution is found. Their method was limited to 3D polyhedral objects with trihedral vertices. In (Stahovich,1995) a stylized sketch along with a user specified description of desired behavior is used to generate a constrained parametric model ensuring behavior. No shape is specified, nor is manufacturing considered.

Sturgill (Sturgill,1995) used sketching of high level mechanical features for early phase CAD design. A traditional menu interface was used for feature creation. Features were displayed with handles that allowed direct manipulation of their geometric parameters. Default parameter values were used if the user did not indicate a preference. Her system could be used for specification and editing of constrained parameters between features and parts. It could also be used to derive process plan information for parts, which possibly

require multiple setups, and to create custom fixtures using the cross part constraints. CNC machining instructions could be generated automatically from the features and process plan representation using the underlying CAD modeler (Alpha_1,1992) at any time. This system did not attempt to do automated manufacturing and was intended primarily as an aid to designers knowledgeable in manufacturing.

Earlier work in this area was done by Anderson and Chang (Anderson,1990) who used a limited set of design level features with handles to form CSG like models. They used a feature classification scheme along with an approximate polyhedral model to determine feature interactions and tool approach directions. Their system produced generative process plans for milling but did not address fixturing.

To automate the manufacturing process it is necessary to analyze the manufacturability of a product and then actually to produce process plans and CNC tool paths. Both of these aspects require extensive geometric computation and reasoning capabilities and research into their automation is ongoing. Approaches have had two distinct thrusts. One is to guarantee that a design is manufacturable as it is created. The other is to create a design and then to evaluate and rate its manufacturability. There is research that exists across the spectrum between these approaches. For a general overview and survey of the state of manufacturability analysis, see (Gupta,1997).

One approach to guaranteeing manufacturability is to design using *manufacturing features*; a parameterized volumetric template that represents the solid volume removed from a workpiece by a single machining operation with a single cutting tool in one tool setup. This clearly has its limits since it embeds a single machine's process within the model. See (Cutkosky,1990; Cutkosky,1992) and more recently (Krishnan,1997).

Post design manufacturability analysis usually is performed on feature based models. Most current CAD systems are either not feature-based or use features that do not contain sufficient information pertinent to this analysis. For this reason there is significant research into feature recognition for manufacturing analysis (Henderson,1984; Gadh,1992; Regli,1994; Regli,1995). Other research decomposes the geometric model into basic removal volumes (Gaines,1995).

Automating process planning is a separate research topic. It is closely related to manufacturability analysis since most analysis approaches require process planing information. This information is either assumed by the analysis package or is generated as a set of alternatives by the package (Gupta,1995).

Process planning involves process selection, determining precedence constraints, including accessibility

(Woo,1994; Elber,1995), fixturability and setup planning, determination of machining operations, tool selection, CNC code generation, feeds and speeds specification, and plan evaluation. There are other considerations for avoiding cutting tool and tool holder interference with the workpiece and fixturing. In (Gupta,1995) *removal* and *accessibility volumes* are used to formalize interference conditions for both the tool and tool holder with the workpiece. Anderson and Chang (Anderson,1990) use a similar approach to determine interference-free access directions for features. Solving problems in each of these topics are major research issues. The interdependencies in these issues makes automating process planning an extremely difficult problem for arbitrary parts. The two most common approaches to computer-aided process planning are variant and generative process planning (Chang,1985; Wang,1991). Variant process planning techniques are used to retrieve and modify existing process plans based on classification schemes for families of parts. Generative process planning uses part features to generate process plans directly.

Automatic toolpath generation, for both roughing and finishing passes, is an important issue in the creation of process plans. This is an area of significant ongoing research in the field. The Sketch-N-Make system requires that CNC code be generated automatically from the high-level mechanical features described in the next section.

APPROACH

In this section we describe our general approach to constructing a prototype *art-to-part* system, from sketching to automated machining of parts.

Broadly, Sketch-N-Make accomplishes a series of transformations from gestures to process plans and CNC programs (see Figure 1). Rather than the customary menu picking-and-pointing use of the mouse or tablet, we use a gestural interface paradigm. An inferencing algorithm recognizes pre-defined patterns of gestures as constructing or modifying geometric shapes. Geometric shapes are automatically transformed into parallel feature-based model and related feature-based process plan representations. These form the bridge between the geometric shapes produced in the user interface and automated manufacturing. Finally, CNC machining programs are algorithmically generated from the feature-based model and process plan representations, and used in automated production of the desired parts (see Figure 2).

Overview of Sketching Graphical User Interface

In this graphical interface paradigm no menus or toolbars are used, so the designer can focus more directly on

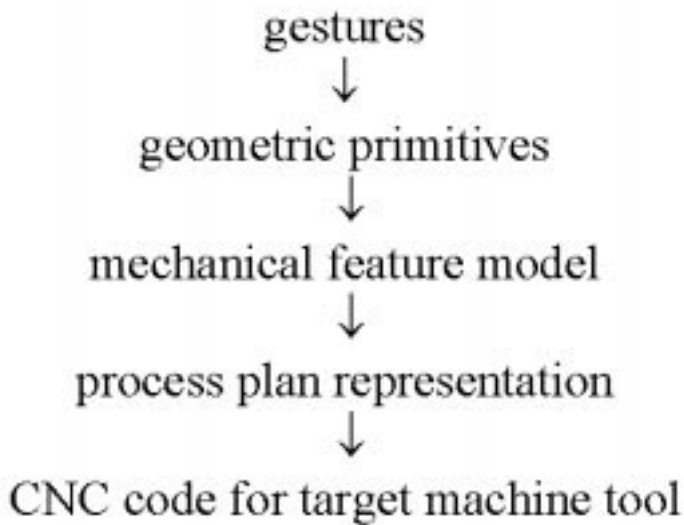


Figure 1. System overview.

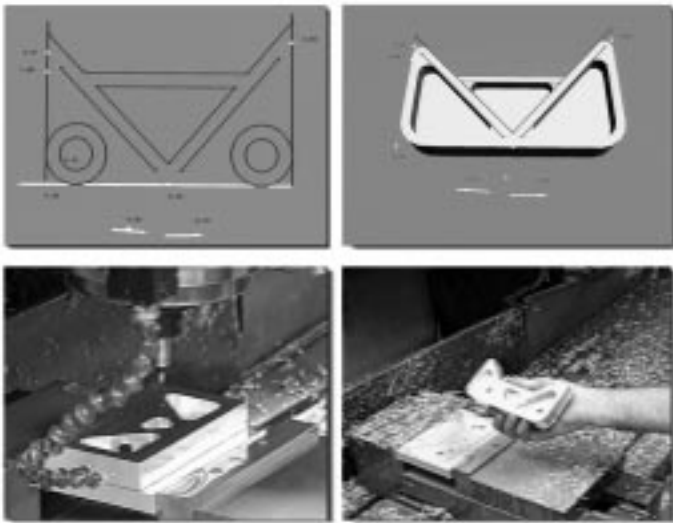


Figure 2. The user creates a 3D geometric model from sketched planar constructions. The geometry is transformed into a feature-based model from which CNC code is derived.

the design itself (Zeleznik,1996). Sketched gestures are geometrically projected from the screen plane onto a scene consisting of an initial ground-plane and, possibly, previously sketched shapes, so the gestures are actually interpreted in a three-dimensional modeling space.

The gesture inferencing algorithm recognizes patterns of gestures and constructs geometric shapes. For example, three straight line gestures, orthogonal from the same starting point on an existing surface, define a block. Two parallel

straight line gestures define a cylinder. Positive volumes are constructed upward, while negative ones, which are to be subtracted away, are constructed downward. Profile curves may be rough-sketched or geometrically constructed from 2D line and circle primitives with parametric dimensioning. These curves can then be used to create complex extruded solids.

Overview of Feature-Based Process Plans

There are two steps in transforming the sketched geometry into a manufacturing description: transforming the primitive geometry into a feature-based model and then creating an appropriate manufacturing process plan template from this model.

Our *mechanical feature objects* represent the design of functional part shapes in the language of mechanical engineering, including, for example, slots, holes, pockets, and grooves (Drake,1989; Cho,1989; Alpha_1,1992). Besides their convenience as high-level constructions for shape models, these feature objects also contain rules to automatically generate fragments of CNC machining programs. Additionally, separate *process plan objects* represent the necessary context to interpret and compose the feature objects into a complete manufacturing process. The process plan objects are divided into *stages* each of which contains fixturing information and the sequence of features to be machined in a single machine set-up.

The term *features* has been used in a variety of ways by various authors. Often it is used to denote a geometric model of the volume of material to be removed by a single cutting tool during a machining process. Our features are higher-level, and more like those described in (Shah,1995; Vieira,1995) in that they contain multiple levels of abstraction. For example, they are higher level in the object-oriented sense, since our feature objects encapsulate both representational data and procedural algorithms which can do considerable geometric computing within a domain context. In this application, the domain context is provided by the library of available cutting tools and the process plan model. Our features are also higher level in the design modularity sense, since a higher-level feature is decomposed into sets of less complex features, which can be further interpreted in context. Our feature set also includes several kinds of patterns replicating an identical lower-level feature to multiple locations (e.g. a bolt-hole circle), as well as parameterized user-defined *product features* which support modular design within product lines. See (Jacobs,1998) for examples from other domains.

During the automatic CNC code generation process, domain knowledge is invoked to transform higher-level features into sets of single-tool features, select cutting tools,

calculate spindle speeds and cutter feed rates, and calculate linked sequences of cutting tool motions. Significant geometric computing support is employed from the underlying object-oriented Alpha_1 CAD/CAM testbed modeling system (University of Utah), including curve, surface, and solid model representations and algorithms based on trimmed NURBS (Non-Uniform knot vector Rational B-Splines.) The sequence of features in the process plan model stages may be considered either as a literal sequence, or processed further in useful ways for optimization, as for example sorting by depth along the common Z orientation. The full process plan feature set for CNC code generation which is supported by our modeler is considerably more general than is utilized in the prototype system described in this paper, and includes features for turning and 5-axis sculpturing (Fish,1998).

Below we describe the novel aspects of the sketching user interface and manufacturing model transformation algorithms in more detail.

SKETCHING USER INTERFACE

The primary goals of the user interface (UI) are: 1) to exploit familiar pencil and paper idioms for conceptual design, 2) provide constructions that fit well with the stereotyped manufacturing process used, and 3) provide mechanisms to constrain and parameterize these constructions in ways required for dimensioned part design and design iteration. We present here a brief overview of the user interface. See (Zelevnik,1996) for more details.

The UI employs an interaction mode that uses gestures which are interpreted relative to view-dependent gesture alphabets for inferring 2D or 3D geometric constructions. A gesture is one or more line strokes drawn with a mouse or tablet stylus. The active gesture alphabet is determined automatically by the viewing angle relative to a reference ground plane.

The user creates 3D geometry from precise planar curve geometry by first creating a network of 2D constructions which are then used to parameterize 3D constructions. Subsequent changes to underlying 2D constructions then propagate through a dependency mechanism to the dependent 3D constructions. Precise numerical relations may be established by entering text values or by using drag and drop techniques which then automatically establish dependencies between dimensioned entities. Gestures support interactive widgets which can be used to change numeric values and explore the design space interactively (Conner,1992).

The 2D geometry mode includes constructions for points, lines, and circles inferred by individual gestures for these entities. Constraints between newly constructed objects and existing ones are inferred automatically. These in-

clude constraints such as point-on-line, line-offset-from-line, and circle-tangent-to-circles among others. All 2D constructions are drawn on a plane (initially the reference ground plane) embedded in the 3D scene.

Currently constructions for 3D geometric primitives are restricted to extrusions of positive or negative volume (negative volumes specify volume removal). Each construction is inferred from a suitable set of individual gestures for points, straight lines, and curved lines. Straight line gestures are typically constrained to be parallel to the principal axes. Such gestures form the basis for constraining constructed 3D geometric primitives to have extrusion axes that are parallel to the Z axis. The approach direction for 3-axis machining operations is taken to be this same direction. Constructions in 3D can be interpreted relative to existing 2D constructions in the scene by gesturing directly above the existing 2D constructions from an oblique view.

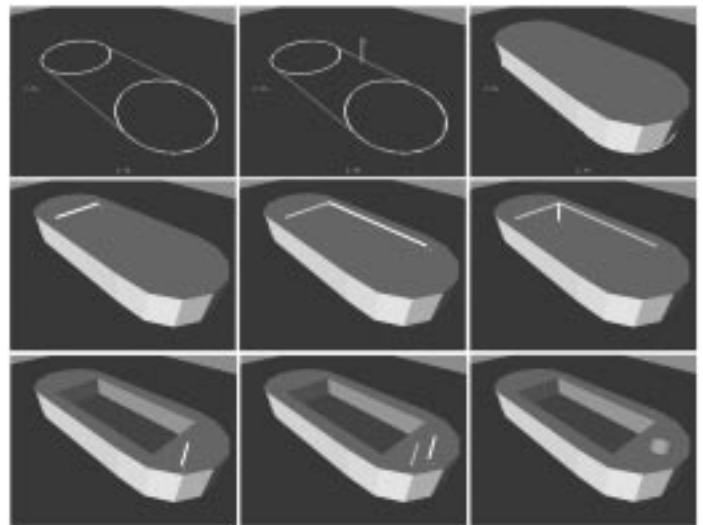


Figure 3. Gestural construction of an extrusion with a pocket and hole. The user first extrudes a 2D profile and then creates a rectangular parallelepiped and cylinder as negative volumes extending into the extrusion.

Figure 3 shows the construction of a 3D extrusion with a pocket and hole. The basic extruded shape is created using a straight line gesture drawn in the positive Z direction above an existing 2D curve. The pocket is formed by drawing two straight lines at right angles to one another on the top surface of the extrusion (giving the X and Y extent of the pocket) and drawing a third line down into the solid (giving the depth of the pocket). The hole is specified by drawing two parallel lines down into the solid.

An abbreviated gesture exists for special holes that are used for fixturing in the manufacturing process. These *fix-*

ture holes are used to accept bolts that both hold and register a part against a fixture plate during machining operations (see **Manufacturing Process Description** below). They are highly constrained in that diameter, length, and threading must match standard commercially available shoulder bolts. They are created with minimal input; the user positions the mouse or tablet's sprite at the point on the part's surface where the hole's center should be located and then presses a designated key.

MACHINED PARTS FROM GEOMETRY

In this section we discuss the process of transforming the geometric shapes produced by the gestural user interface into mechanical feature-based process plans that will be used to drive the manufacturing process. We then discuss the manufacturing processes used to actually produce Sketch-N-Make parts.

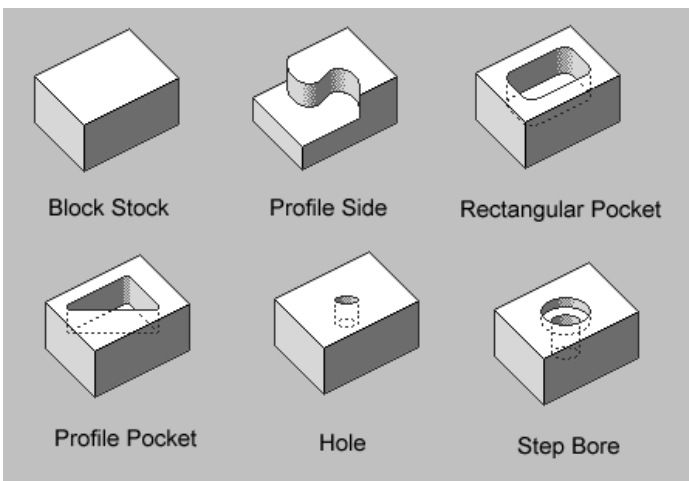


Figure 4. Mechanical features currently used in transformation process.

A Mechanical Feature Vocabulary

The mechanical features currently used in the transformation process are depicted in Figure 4. These features include detailed geometric and manufacturing information **not** present in the geometric primitives created by the user interface. This additional information is provided by a set of defaults, conforming to standard manufacturing rules, which may be changed by advanced users. Enough domain knowledge is present in the mechanical features to derive, within the context of a process plan representation, machining operations for manufacturing.

The features chosen for inclusion in Sketch-N-Make, and itemized below, are a subset of a much larger more inclusive collection supported by the underlying CAD/CAM modeling system.

block stock feature: We will be machining parts from rectangular blocks of metal or other machinable materials such as plastic. A block stock feature represents the position and dimensions of this block. This feature is used as a reference when decomposing other features into machining operations within a stage of the process plan. For example, cutting operations for a profile side (see below) should not extend beyond the boundaries of the block stock.

profile side feature: Gives the outer profile shape for an extruded part and the depth of the extrusion. The milling operations derived from this feature will remove the material of the block stock that is outside the specified profile curve and to the specified depth. The top of a profile side may optionally be chamfered.

rectangular pocket feature: A rectangular pocket is used to remove material from the face of the block stock to a specified depth. A pocket has flat sides, possibly at a specified draft angle, and a flat bottom. A radius may be inserted between the sides and the bottom and the pocket's top portion may be chamfered. The shape of the pocket is specified by the width of the rectangle in both X and Y . The corners of this rectangle are rounded by a specified radius.

profile pocket feature: A profile pocket is more general than a rectangular pocket. Its outer shape is specified by a closed planar curve.

hole feature: A hole feature corresponds to the subtraction of a surface of revolution which will be machined primarily by a drill. Holes may be chamfered, have different diameters along their lengths, and have tapped sections. If a fine tolerance attribute is applied to the hole it will be drilled undersized and then accurately reamed. The diameters of holes are discretized to match available drill and ream sizes.

bore feature: A bore feature is essentially a circular pocket. The manner in which it is machined requires that it be larger than a minimum diameter (which depends on tooling) but it need not have its diameter discretized.

Subtractive features may either be *through features*, extending entirely through the part block, or *blind features*, extending only part way through the part block.

Transformation Algorithms

As the user develops the sketched model, the system tracks and transforms the geometry created in the user in-

terface into a parallel feature-based representation. The system keeps this feature-based representation synchronized with the user interface geometry as the user adds to or modifies the geometric model. The mechanical features in this representation are placed within the context of an overall process plan. We briefly outline this transformation process.

positive volume extrusions: A positive volume extrusion that rests on the ground plane is converted in one of two ways. If the extrusion is a rectangular parallelepiped then it will be converted into a set of features consisting of a block stock for the part and optionally a block stock for a fixture plate. A more general extrusion is converted into a set of features as above but with the addition of a profile side feature for milling the outer shape of the extrusion.

The system derives the X and Y dimensions of both the part and fixture plate block stocks from the bounding box of the extruded part. The height of the block stock for the part is determined by the height of the extrusion. A fixture plate is required for only one of the process plan templates. If the fixture plate block stock is required, it will have a default thickness.

negative volume extrusions: Non-cylindrical, negative volume extrusions are converted into pocket features. If the negative extrusion is a rectangular parallelepiped then it is converted into a rectangular pocket feature. Otherwise it is converted into a profile pocket feature.

negative volume cylinders: Negative volume cylinders are converted into hole or bore features. Cylinders of small enough diameter are converted into hole features. The diameters for these features are discretized to match available tooling using a table lookup mechanism. Cylinders with diameters larger than a preset amount are converted into bore features with no discretization of their diameters.

Negative volume cylinders constructed in the user interface from gestures for fixture holes (see the **Sketching User Interface** section above) are transformed differently. Fixture holes are used to accommodate standard shoulder bolts that both register and hold the part against the fixture plate in stage 4 of process plan template B (see below). These holes are designated as input to the transformation process by negative cylinders that are flagged with a “fixture” attribute. Each cylinder of this form is converted into the following matched set of features:

- A through hole feature in the part block. The hole will be of a preset diameter corresponding to the diameter of the shoulder bolts to be used.
- A counter drilled tapped hole feature in the fix-

ture block. The length of the counter-drilled section is discretized so that standard length shoulder bolts may be used. This is accomplished by table lookup.

Partially Evaluated Process Plans. The actual manufacturing process plan template that will be used to machine a part depends on the fixturing strategy for that part (as discussed in the **Manufacturing Process Description** section below). The fixturing strategy is unknown initially. For this reason the mechanical features that result from the transformation algorithms are initially placed within the context of a *partially evaluated* process plan. This currently consists of a CSG tree with the mechanical features as its nodes.

The mechanical features in the tree can be displayed selectively so that the user can see the **feature** representation of the part being designed (as opposed to the normal user interface geometry).

Once sufficient information is available to determine the fixturing strategy for the part, the system makes a pass over the CSG tree to produce the actual process plan according to the process plan templates described below.

Results of the Transformation Process. The final results of the transformation process are:

- A feature-based model and its related process plan representation which contains a set of stages, each including an ordered set of mechanical features to be manufactured in that stage. The stages of the process plan representation correspond to the stages of the process plan template to be used in the actual manufacturing (these templates are discussed below). The mechanical features included for a given stage are used to automatically derive both the tooling and CNC code used to drive the machining center. Transformations from collections of high-level mechanical features to derived machining operations are themselves the subject of research.
- A *parts list* that gives the dimensions of the part block and, if required, the dimensions of the fixture plate and the size (diameter and shoulder lengths) of the fixture plate shoulder bolts.

Additional Information. The mechanical features contain information not present in the geometric primitives manipulated in the user interface. This information includes: allowed tolerances, chamfer depths, pocket wall draft angles, radii for filleting sharp corners, and radii for pocket

bottoms. The process plan representation also contains additional manufacturing information such as material properties and surface finish requirements.

Values for the above information are set from intelligent defaults which conform to standard manufacturing rules. These defaults may be changed by advanced users.

Manufacturing Process Description

All parts designed using Sketch-N-Make are manufactured using a family of related process plans which we refer to as *process plan templates*. Each template consists of up to four stages and uses at most one custom fixture component. Each of the different templates supports a different group of fixturing strategies which in turn allows for a broader class of parts to be manufactured using the system.

Template A is used for parts that have rectangular footprints, or outlines. Templates B and C represent alternative approaches for fixturing parts that have non-rectangular outlines. The use of template B requires the user to specify the location of at least two bolt holes that will be used to accommodate shoulder bolts that affix the part block to a fixture plate. These locations are specified by simple gestures (as discussed in the **Sketching User Interface** section above). The use of template C requires that the part outline have parallel extremal sides of suitable length. The choice of template is based on the overall part shape and presence or absence of fixture bolt holes specified by the user.

The CNC code that drives the process in all stages (except stage 1) is automatically generated from the feature-based model and process plan representation.

Template A: This process plan is used for parts with rectangular outlines (see Figure 5).

Stage 1: Prepare part block

The part block is first rough cut and then accurately planed to the dimensions given in the parts list.

Stage 2: Cut interior details

The part block is clamped by two of its faces in a vise which is mounted to the table of the machining center. Interior details of the part are then cut. These details include pockets, holes, and bores, but do not include any cutting operations that may interfere with holding the part in the vise.

Template B: This template is one alternative for manufacturing parts that have non-rectangular outlines. Template B consists of stages 1 and 2 from template A with the addition of the following two stages (see Figure 5):

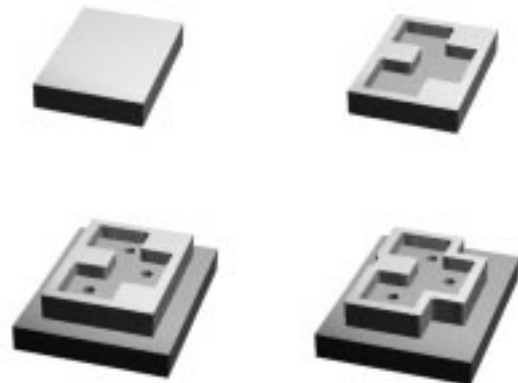


Figure 5. Stages for process plan templates. Top depicts stages 1 and 2 of template A. Bottom depicts stages 3 and 4 of template B.

Stage 3: Prepare fixture plate

A fixture plate is made in this stage which will be used to hold the results of stage 2 during the execution of stage 4. The fixture plate is a block of preset thickness with two or more counter-drilled tapped holes for accepting the shoulder bolts needed to mount the part block in the next stage.

Since the fixture plate model is a rectangular block with only interior details, it can be manufactured using template A.

Stage 4: Cut exterior profile

The fixture plate is clamped to the vise. The part block from stage 2 is then bolted to the fixture plate and the external profile cuts are made to the part block. The fixture plate is used in this stage to raise the part block sufficiently to prevent these cuts from interfering with the vise.

Template C: An alternative fixturing approach is provided for parts with non-rectangular outlines when fixture bolts are not specified. This approach involves automatically extending the thickness of the part block feature model so that the block can be held in the vise by its extended portion while the part profile is being cut. The part will then be turned over and the extension removed. This assumes that the profile of the part has two parallel extremal sides of sufficient length for clamping in the vise during this final operation.

This template consists of stage 1 from template A with the addition of the following two stages:

Stage 2: Cut interior details and exterior profile

The part block is clamped to the vise using only the bottom extended portion of the block. All pocket, hole, and bore features are cut. Then the external profile cuts are made to the outside of the part. The bottom extension of the block is used to prevent any of these cuts from interfering with the vise.

Stage 3: Plane off extension

The part block is turned over and clamped in the vise by two parallel faces. The block extension is then planed off from the bottom of the part.

If more than one copy of a part is being made then individual stages can be repeated as necessary for each part block before moving on to the next stage.

DESIGN RULES

In this section we briefly discuss design rules needed to guarantee the manufacturability of models using our process plan templates. We want to give the user as much flexibility as possible during the design process and hence we allow the user to create intermediate states of the model that are **not** manufacturable. Queries regarding the manufacturability of the model can be initiated at any point in the design process, however. Currently in our prototype Sketch-N-Make system not all of the design rules discussed here are checked automatically. Additional validation procedures are under development. We emphasize that these are rules to check the manufacturability, but **not** the functionality, of parts being designed.

We currently restrict designs to prismatic parts with machining access from the positive Z direction. This restriction is enforced through the allowed gesture and geometric primitive sets of the user interface and verification by the algorithms which transform the geometric primitives to feature-based models. As a further accessibility constraint, visibility of surfaces from the positive Z direction can be analyzed (Elber,1995).

Part dimensions may be restricted by the working volume of the machining center used (determined by spindle and table travel as well as the size of the vise). The available thickness of metal or plastic materials used for stock may also place practical restrictions on the dimensions of a part.

Other dimension rules, such as the length-to-diameter (L/D) ratios for milling cutters and drills, are an important consideration. This is especially true since designers unfamiliar with manufacturing practices will tend to be unaware of the practical restrictions that result from these rules. System users are warned when the combination of cut depth and inside radii of features require tooling that is

either unavailable or outside common machining practice. Users are also warned when hole diameters are discretized away from their original specifications because of available tooling for drills and reams.

Also important are design rules which guarantee that the mill will not interfere with the fixturing components used in the process plan templates. For example, a simple rule that prevents interference with the vise requires that features cut in templates A and B to Z depths at or below the surface of the vise must not intersect, or extend beyond, the sides of the part block used for clamping. Similarly, another rule requires that through features cut using templates A or B must not be within a system preset distance of these sides. This is to prevent interference with the portion of the vise that supports the part block from below. Also, a third rule requires that cutting operations do not intersect the shoulder bolts used in template B and that the bolt heads have ample clearance if they are placed in pocket bottoms. The distances of centers for bolt holes from the closed profile side and profile pocket curves serve as one check for bolt clearance problems.

It is possible that deflection and deformation of the workpiece might be caused by torques on inadequately placed fixture bolts used with process plan template B. This may be checked by conservative "rules of thumb" which relate overall part dimension, depth of feature cuts, material hardness, and distance between, and placement of, the fixturing bolts.

CONCLUSIONS AND FUTURE WORK

Our prototype art-to-part system furnishes a smooth path from early concept design through detailing to automated manufacture of machined parts. It demonstrates that suitably restricted and standardized classes of parts and set-ups coupled with a sketching user interface front end provide an environment and procedure for extremely rapid part creation and production. This system allows a user without manufacturing expertise to concentrate on geometric and parametric aspects of design without specifying details required for the manufacturing process. Only the machine tool and machine operator are needed to complete the part's manufacture.

The advantages of this system have been achieved by identifying a combination of carefully matched elements: restricted part domain, a sketching user interface paradigm incorporating a well chosen set of gestures, the set of 3D geometric shapes produced by this interface, the transformation of these shapes into selected high-level mechanical features, and the process plan templates which provide context for these features and represent the family of manufacturing processes selected. The proper matching of these elements

allows for significant automation in the system and hence a speed up of the overall art-to-part process. The gestural user interface is also well matched to the cognitive abilities of users, exploiting familiar pencil and paper sketching paradigms. This helps speed up the input process.

Another important aspect of the system's design is that it maintains two levels of abstraction from the point of view of manufacturing. Although users can manipulate detailed manufacturing information if they desire, they are not forced to do so. If a user chooses to manipulate these details they are not forced to **continue** to do so subsequently. A fully detailed feature-based model and process plan representation are automatically maintained in synchrony with the geometric model as it is created and changed. The user need not be concerned with the manufacturing details. From this standpoint, the geometric shapes of the model become abstract handles for both the mechanical feature-based and process plan representations.

Even in its prototype stage we feel that Sketch-N-Make demonstrates a useful and practical tool for designers. The system currently works with a class of non-trivial prismatic parts that have wide practical use. Further, the manufacturing requirements of the process plan templates are fairly standard. Although these process plans do not represent optimal overall strategies for manufacturing parts, individual stages of the manufacturing process are optimized (to minimize tool changes during that stage, for example). We believe that this approach is currently viable for the production of prototypes, small lots, and tooling for larger production runs. These are cases where optimal overall strategies are less important.

Our future work in this area will be devoted to expanding this approach and trying to establish the limits to which it can naturally be extended. We feel it can be taken quite far before fundamental difficulties in the paradigm arise.

We are particularly interested in extending the approach to wider classes of parts. Extending the process plan templates to allow more fixturing strategies and different workpiece orientations (for different tool approach directions) is one area of investigation. Other areas include extending the approach to work with different processes (such as turning) and assemblies of parts.

We also intend to investigate how far the sketching gestural interface paradigm can be taken in specifying increasingly complex CAD designs. This will require extended gesture sets that can be interpreted as high level 2D and 3D constructions. Closely related to extending the gesture sets is the use of a larger class of high level mechanical features in the system such as, for example, additive protrusion features.

ACKNOWLEDGMENT

This work was supported in part by the NSF Science and Technology Center for Computer Graphics and Scientific Visualization (ASC-89-20219) and DARPA (F33615-96-C-5621) with additional support from Alias/Wavefront, Advanced Networks and Services, Autodesk, Microsoft, Sun Microsystems, and TACO Inc. All opinions, findings, conclusions or recommendations expressed in this document are those of the authors and do not necessarily reflect the views of the sponsoring agencies.

REFERENCES

- Alpha_1 Research Group, University of Utah. *Alpha_1 User's Manual*, 1992.
- D.C. Anderson and T.C. Chang. Geometric reasoning in feature-based design and process planning. *Computers & Graphics*, 14(2):225–235, 1990.
- Tien-Chien Chang and Richard A. Wysk. *An Introduction to Automated Process Planning Systems*. Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1985.
- Byung Sung Cho. Rule based process planning system for hole feature machining. Master's thesis, University of Utah, Department of Computer Science, August 1989.
- D.B. Conner, S.S. Snibbe, K.P. Herndon, D.C. Robbins, R.C. Zeleznik, and A. van Dam. Three-dimensional widgets. *Computer Graphics (Proceedings of the 1992 Symposium on Interactive 3D Graphics)*, 25(2):183–188, March 1992.
- M. Cutkosky and J. Tenenbaum. A methodology and computational framework for concurrent product and process design. *Mech. Mach. Theory*, 25(3):365–281, 1990.
- M. R. Cutkosky and J. M. Tenenbaum. Toward a framework for concurrent design. *International Journal of Systems Automation: Research and Applications*, 1(3):239–261, 1992.
- Samuel Drake and Samuel Sela. A foundation for features. *Mechanical Engineering*, 111(1):66–73, January 1989.
- Lynn Egli, Beat D. Bruderlin, and Gershon Elber. Sketching as a solid modeling tool. *Third Symposium on Solid Modeling and Applications (Proceedings)*, pages 313–321, 1995.
- G. Elber and E. Cohen. Arbitrarily precise computation of gauss maps and visibility sets for freeform surfaces. *Proceedings of the Third ACM/IEEE Symposium on Solid Modeling and CAD/CAM Applications*, 1995.
- Russell Fish, Samuel Drake, Elaine Cohen, and Richard Riesenfeld. Feature-based process planning for cnc machining. Technical Report UUCS-98-001, University of Utah, Department of Computer Science, Salt Lake City, Utah, 1998.
- R. Gadh and F. B. Prinz. Recognition of geometric

forms using the differential depth filter. *Computer Aided Design*, 24(11), November 1992.

D. Gaines, C. Hayes, and Y. Kim. Negative volume to process methods mapping for cad/capp integration. *Proceedings of IJCAI Intelligent Manufacturing Workshop*, 1995.

I. Grimstead and R. Martin. Creating solid models from single 2d sketches. *Proceedings of the Third Symposium on Solid Modeling and Applications*, pages 323–337, 1995.

S. K. Gupta, D. Das, W. C. Regli, and D. Nau. Automated manufacturability analysis: A survey. *Research in Engineering Design*, 9(3), 1997.

S. K. Gupta and D. Nau. Systematic approach to analysing the manufacturability of machined parts. *Computer-Aided Design*, 27(5):323–342, 1995.

Mark R. Henderson. *Extraction of Feature Information from Three-Dimensional CAD Data*. PhD thesis, Purdue University, West Lafayette, IN, 1984.

T. Jacobs and E. Cohen. Aggregation and controlled interaction: Automated mechanisms for managing design complexity. *Proceedings of the 1998 ASME 10th International Conference on Design Theory and Methodology*, 1998.

S. Krishnan and E. Magrab. An integrated dfm system for milling. *Proceedings of the 1997 ASME Design for Manufacturing Conference*, 1997.

W. Regli, S. Gupta, and D. Nau. Interactive feature recognition using multi-processor methods. *Proceedings of the 1995 ASME Computer Integrated Design Conference*, 1995.

William C. Regli, Satyandra K. Gupta, and Dana S. Nau. Feature recognition for manufacturability analysis. *Proceedings of the 1994 ASME Computers in Engineering Conference*, 1994.

Jami J. Shah and Martti Mäntylä. *Parametric and Feature-Based CAD/CAM*. John Wiley & Sons, Inc., 1995.

F. Stahovich, R. Davis, and H. Shrobe. Turning sketches into working geometry. *Proceedings of the 1995 ASME 9th International Conference on Design Theory and Methodology*, pages 603–610, 1995.

M. Sturgill, E. Cohen, and R. F. Riesenfeld. Feature based 3-d sketching for early design. *Proceedings of the 1995 ASME Computers in Engineering Conference*, 1995.

A. Vieira. Consistency management in feature based parametric design. *Proceedings of the 1995 ASME 9th International Conference on Design Theory and Methodology*, pages 977–987, 1995.

H. Wang and J. Li. *Computer-Aided Process Planning*, volume 13 of *Advances in Industrial Engineering*. Elsevier, 1991.

T. Woo. Visibility maps and spherical algorithms. *CAD*, 26(1), 1994.

Robert C. Zeleznik, Kenneth P. Herndon, and John F.

Hughes. Sketch: An interface for sketching 3d scenes. *Computer Graphics (SIGGRAPH '96 Proceedings)*, pages 163–170, August 1996.