AI Planning Versus Manufacturing-Operation Planning: A Case Study[*]

Dana S. Nau[†]
Computer Science Department and
Institute for Systems Research
University of Maryland
College Park, MD 20742
nau@cs.umd.edu

Satyandra K. Gupta
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
skgupta@isl1.ri.cmu.edu

William C. Regli[‡]
Computer Science Department and
Institute for Systems Research
University of Maryland
College Park, MD 20742
regli@cs.umd.edu

Abstract

Although AI planning techniques can potentially be useful in several manufacturing domains, this potential remains largely unrealized. In order to adapt AI planning techniques to manufacturing, it is important to develop more realistic and robust ways to address issues important to manufacturing engineers. Furthermore, by investigating such issues, AI researchers may be able to discover principles that are relevant for AI planning in general. As an example, in this paper we describe the techniques for manufacturing-operation planning used in IMACS (Interactive Manufacturability Analysis and Critiquing System), and compare and contrast them with the techniques used in classical AI planning systems. We describe how one of IMACS's planning techniques may be useful for AI planning in general—and as an example, we describe how it helps to explain a puzzling complexity result in AI planning.

1 Introduction

AI planning techniques can potentially be useful in several manufacturing domains. However, with the exception of manufacturing scheduling, previous applications of AI planning technology to manufacturing (cf. [Famili *et al.*, 1992]) generally have had little impact on manufacturing practices [Ham and Lu, 1988; Nevins and Whitney, 1989; Shah *et al.*, 1994].

One reason for this difficulty appears to be the different world views of AI planning researchers and manufacturing planning researchers. The first author works in both worlds—and his work on manufacturing planning has significantly influenced his research on AI planning, and vice versa—but this influence is not particularly evident in the publications themselves, because they were written to address two different audiences, who have different ideas of what the important problems are and how they should be solved:

- Since *AI planning researchers* are usually more interested in general conceptual problems than domain-dependent details, the AI approach to manufacturing planning has typically been to create an abstract problem representation that omits unimportant details, and look for ways to solve the abstract problem. From the viewpoint of the manufacturing engineer, these "unimportant details" often are very important parts of the problem to be solved—and this can lead manufacturing engineers to view AI planning techniques as impractical.
- *Manufacturing planning researchers* typically want to solve a particular manufacturing problem, and present their research results within the context of this problem, without discussing how the approach might generalize to other planning domains. For AI researchers, this makes it difficult to see what the underlying conceptual problems are, or whether the approach embodies a general idea that can be applied to other problems. This can lead AI planning researchers to view manufacturing planning as a domain full of ad-hoc, domain-specific programs rather than general principles and approaches.

Some of the issues arising in manufacturing planning are similar to issues investigated in AI planning, and others are distinctly different. Some of the former may amenable to the use of existing AI planning techniques—and some of the latter may lead to new principles useful in AI planning. However, to investigate such issues, AI researchers will need a better understanding of manufacturing problems and concerns, so as to get better ideas of what the interesting generalizations are, and which techniques from AI might best be applied to realistic manufacturing problems.

In this paper we attempt to provide a step in this direction, by describing the planning techniques used in IMACS, a computer system for helping designers produce designs that are easier to manufacture [S. Gupta *et al.*, 1994b; S. Gupta and Nau, 1995]. IMACS analyzes the manufacturability of proposed designs for machined parts by generating and evaluating operation plans for

[*] This work was supported in part by NSF Grants DDM-9201779, IRI-9306580, and NSFD EEC 94-02384. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

[†] Also with the University of Maryland Institute for Advanced Computer Studies (UMIACS).

[‡] Also with: National Institute of Standards and Technology, Manufacturing Systems Integration Division, Building 220, Room A-127, Gaithersburg, MD 20899.

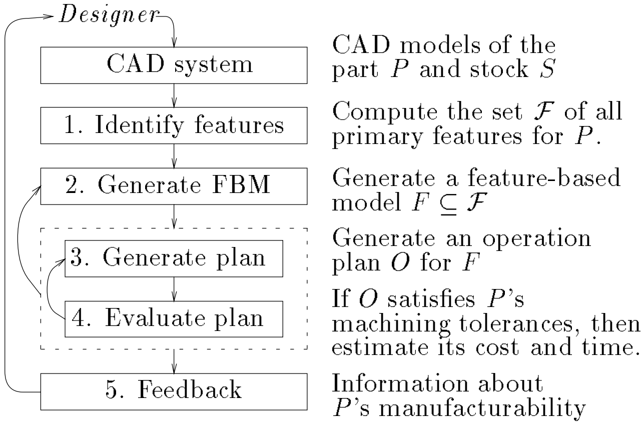

Designer
CAD system
CAD models of the part P and stock S
1. Identify features
Compute the set $\mathcal{F}$ of all primary features for P.
2. Generate FBM
Generate a feature-based model $F \subseteq \mathcal{F}$
3. Generate plan
Generate an operation plan O for F
4. Evaluate plan
If O satisfies P's machining tolerances, then estimate its cost and time.
5. Feedback
Information about P's manufacturability

Figure 1: Basic approach used in IMACS.

the proposed design. We discuss similarities and differences between the techniques used in IMACS and those typically used in AI planning. We also describe how one of IMACS's planning techniques (the enumeration of relevant tasks before planning begins) may be useful for AI planning in general—and as an example, we describe how it helps to explain a puzzling complexity result in AI planning.

2 A Case Study: IMACS

IMACS (Interactive Manufacturability Analysis and Critiquing System) is a computer system for analyzing the manufacturability of machined parts, in order to help designers produce designs that are easier to manufacture. Further information about IMACS, including color images produced using it, are available at http://www.cs.umd.edu/projects/cim/imacs/imacs.html.

As shown in Figure 1, IMACS evaluates the manufacturability of a proposed design by generating and evaluating operation plans. Here are two immediate differences between IMACS and many AI planning systems:

- Unlike most AI planners, IMACS generates more than one plan and evaluates the merit of each plan it generates, to find an optimal plan. To measure plan merit, IMACS uses an estimate of the plan's manufacturing time, as described in Section 2.6. However, it is straightforward to incorporate estimates of production cost as well [S. Gupta *et al.*, 1994c].
- We are developing ways for IMACS to suggest changes in the design to improve its manufacturability while still fulfilling the designer's intent [Das *et al.*, 1994; 1995]. In AI terms, this means automatically suggesting changes to the goal to make it easier to achieve.

Other differences and similarities are discussed in the following sections.

2.1 Machined Parts

A machined *part*, P, is the final component created by executing a set of machining operations on a piece of *stock*, S. For example, Figure 2 shows a socket P_0, and the stock S_0 from which P_0 is to be produced. Note that the goal to be achieved (i.e., the part to be produced) is represented not as a set of predicates as is often done in AI planners, but instead as a CAD model (which IMACS represents using ACIS, a solid modeling system from Spatial Technologies Inc.).

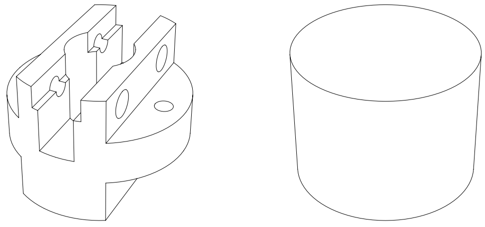


Figure 2: The socket P_0 and the stock S_0.

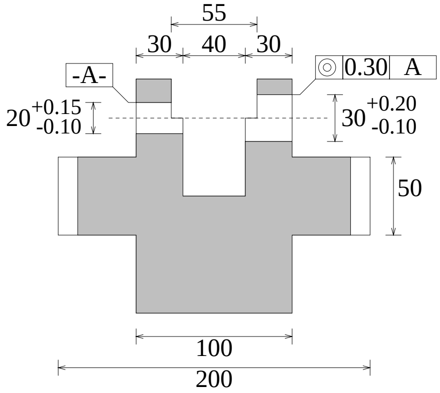
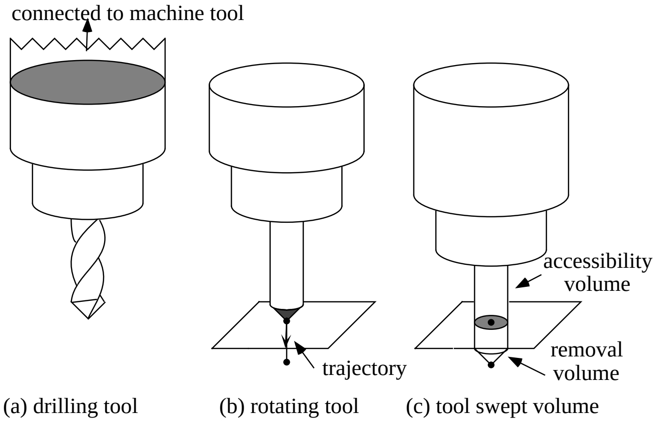


Figure 3: Dimensions and tolerances for the socket P_0



(a) drilling tool (b) rotating tool (c) tool swept volume

Figure 4: Example of a machining operation.

An *operation plan* is a sequence of machining operations capable of creating the part P from the stock S. Since it would be physically impossible to produce P's *exact* geometry, designers give *design tolerance* specifications (e.g., see Figure 3) to specify how much variation from the nominal geometry is allowable in any physical realization of P. A plan is considered capable of achieving the goal if it can create an approximation of P that satisfies the design tolerances.

A *workpiece* is the intermediate object produced by starting with S and performing zero or more machining operations. Currently, the machining operations considered in IMACS include end milling, side milling, face milling and drilling operations, on a three-axis vertical machining center. Each machining operation creates a *machining feature*. Different researchers use different definitions of machining features; as shown in Figure 4, we consider a machining feature to include information about the type of machining operation, the material removal volume (the volume of space in which material can be removed), and the accessibility volume (the volume of space needed for access to the part).

2.2 Feature Extraction

Although much past work on integrating design with manufacturing planning has involved *feature-based design* techniques in which users specified designs directly as sets of form features, most researchers have become convinced that a single set of features cannot satisfy the requirements of both design and process planning—instead, some form of feature extraction is needed. For IMACS, we have developed algorithms to extract machining features directly from the CAD model [Regli *et al.*, 1994; S. Gupta *et al.*, 1994a].

There can be many—sometimes infinitely many—different machining features capable of creating various portions of a given part. Of these, we define a *primary* feature to be a feature that contains as much of the stock as possible without intersecting with the part, and as little space as possible outside the stock. Figure 5 shows examples of primary and non-primary features; for a detailed definition see [S. Gupta and Nau, 1995].

As described in [S. Gupta *et al.*, 1995; Regli *et al.*, 1995], in every operation plan that IMACS will ever want to consider, each machining operation will create either a primary feature or a truncation of a primary feature—and the number of primary features for a part is always finite (in fact, polynomial). Thus, IMACS's first step is to find the set $\mathcal{F}$ of all primary features for P and S. For example, for the socket P_0 the set $\mathcal{F}$ contains 22 primary features, a few of which are shown in Figure 6.

In AI terms, machining operations are elementary actions and machining features are tasks. $\mathcal{F}$ is the set of all tasks that might ever be relevant for achieving the goal. Unlike most AI planners, IMACS finds this set in advance before it begins to generate plans—but as we discuss later, this technique may be useful in a number of AI planning problems.

2.3 Generating Incomplete Plans

Figure 6 shows that the features in $\mathcal{F}$ may overlap in complicated ways, and not all of them are needed to create the part (for example, we do not need to machine both $s1$ and $s2$). A *feature-based model* (FBM) is any irredundant subset of features $F \subseteq \mathcal{F}$ such that subtracting those features from S produces P. For example, Figure 7 shows an FBM, FBM1, for the socket P_0.

In AI planning terminology, an FBM is an incomplete plan: if we can machine the features in it, this will create the part. Since each FBM is a subset of $\mathcal{F}$, FBM's can be generated using set-covering techniques, but there can be exponentially many FBM's. As an example, for the socket P_0, $\mathcal{F}$ contains 22 primary features from which one can form 512 FBM's. In general, we usually will not want to generate *all* of these FBM's, for only a few of them will lead to good operation plans. Thus IMACS does a depth-first branch-and-bound search to generate and test FBM's one at a time, pruning unpromising

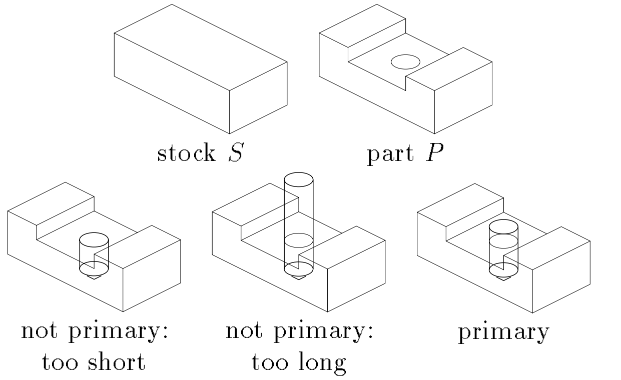


Figure 5: Non-primary and primary drilling features.

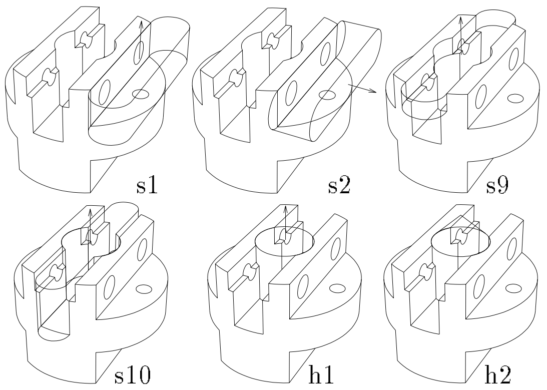


Figure 6: A few of the 22 primary features for the socket P_0. $s1$, $s2$, $s9$, and $s10$ are end-milling features; $h1$ and $h2$ are drilling features.

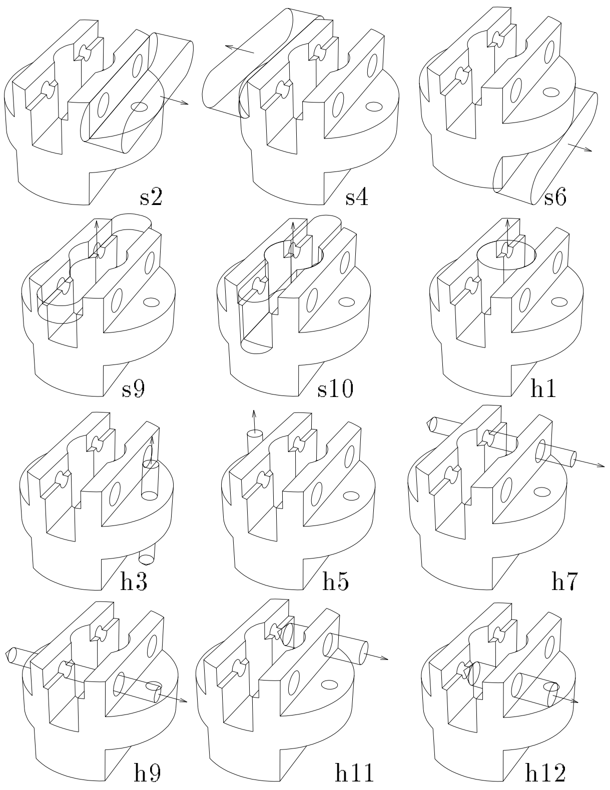
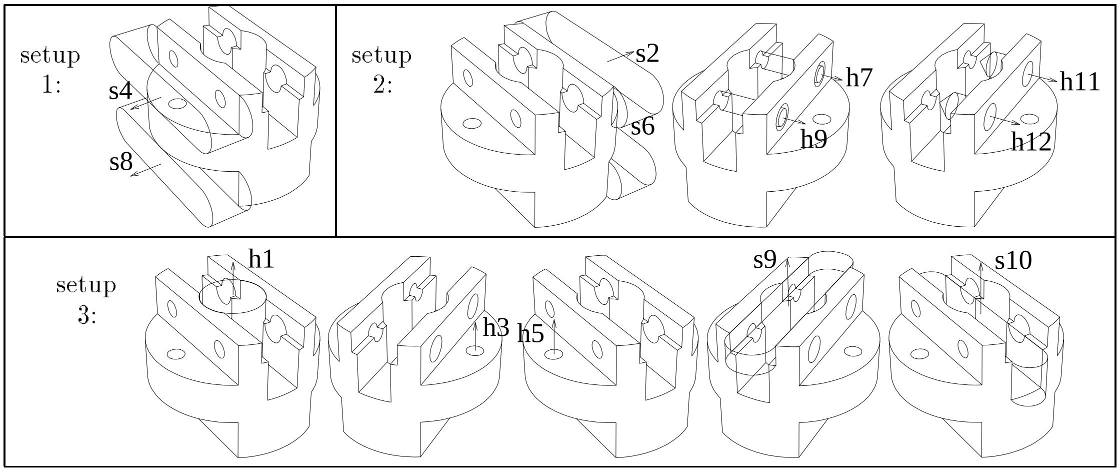
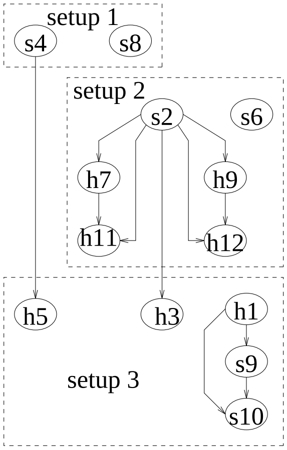


Figure 7: Feature-based model FBM1 for the socket P_0.



(a) features to be machined

(b) ordering constraints



(c) process details

Feature name	Feature type	Tool diam (mm)	Feed rate (mm/min)	Number of passes	Pass length (mm)
$s4$	end-milling	50	166	2	225
$s8$	end-milling	50	166	2	225
$s2$	end-milling	50	166	2	225
$s6$	end-milling	50	166	2	225
$h7$	drilling	20	244	1	106
$h9$	drilling	20	244	1	106
$h11$	drilling	30	203	1	39
$h12$	drilling	30	203	1	39
$h1$	drilling	75	108	1	172.5
$h3$	drilling	20	244	1	56
$h5$	drilling	20	244	1	56
$s9$	end-milling	50	166	1	250
$s10$	end-milling	40	207	3	240

Figure 8: An operation plan derived from FBM1. This plan is the optimal one for making P_0. Note that each feature is either a primary feature from FBM1 or a truncation of a primary feature from FBM1.

FBM's as described in Section 2.7. For example, IMACS generates only 16 of the 512 FBM's for the socket P_0.

In many of the early generative process planning systems (e.g., [Chang and Wysk, 1985; Nau and Chang, 1986; Nau, 1987]), the input was a symbolic representation of P as a set of machining features analogous to a single FBM, with no way to recognize or handle many of the geometric interactions among the features. This prevented such systems from generating realistic process plans for complex parts, in which geometric interactions can make it quite difficult to decide what sets of features and machining operations to use, which operations to do when and in which setups, and how to hold the workpiece during each setup.

In one way or another, most recent work on generative process planning (both by manufacturing researchers and AI researchers) has tried to address these difficulties (e.g., [Kambhampati *et al.*, 1992; Vandenbrande and Requicha, 1993; Opas and Mäntylä, 1994; S. Gupta *et al.*, 1994b; Das *et al.*, 1994; Hayes, 1995; Britanik and Marefat, 1995]). However, there are also some recent AI efforts at process planning that unfortunately do not seem to address such difficulties at all. We suspect one reason for this is that the researchers involved in these efforts lack sufficient familiarity with the problem domain—and Section 4 describes a way whereby we hope to alleviate this problem.

2.4 Resolving Goal Interactions

An FBM is basically a totally unordered plan. To resolve goal interactions, IMACS adds ordering constraints as follows:

- *Identify ordering constraints.* Due to complex geometric interactions (accessibility etc.), some features must precede others. For example, in Figure 8, the hole $h1$ must be machined before the slot $s9$ in order to achieve reasonable machining tolerances and avoid tool breakage.
- *Linearize.* Next IMACS generates all total orderings consistent with the precedences. If no such total ordering can be found, IMACS considers the FBM F to be unmachinable and discards it. Unlike the typical approaches used in AI planners, there would be no point in adding additional operators: they would just create redundant features, and if there is a feasible way to machine the part it will be found among the other FBM's.

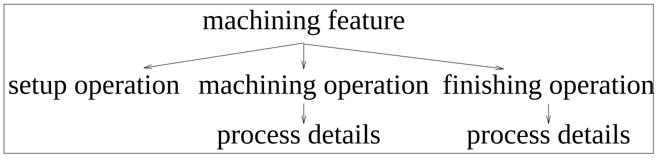


Figure 9: Task decomposition in IMACS.

- *Modify goals.* Suppose features f and g overlap, and f precedes g in some total ordering. Then when we machine f, we are also machining part of g. We don't want to machine that same portion of g again later in the sequence, because we would merely be machining air. Thus, IMACS truncates g to remove the portion covered by f. As an example, several of the features shown in Figure 8(a) were produced by truncating the corresponding features in FBM1.
- *Unlinearize.* Once the truncated features have been produced, several of the resulting FBM's may have identical features but different precedence constraints. In such cases the precedence constraints that differ can be removed, translating the total orders into partial orders. For example, Figure 8(b) shows the partial order for the FBM of Figure 8(a).

2.5 Additional Steps

To obtain an operation plan from the partially-ordered FBM, IMACS uses the following steps:

- *Incorporate finishing operations.* For faces with tight surface finishes or tolerances, IMACS adds finishing operations, with precedence constraints to make them come after the corresponding roughing operations. Currently, one finishing operation per face is allowed.
- *Determine setups.* On a three-axis vertical machining center, features cannot be machined in the same setup unless they have the same approach direction. This and the partial ordering constraints can be used to determine which features can be machined in the same setup, as shown in Figure 8(b). Although the specific computations are different, the problem is a special case of what is known to AI researchers as the plan-merging problem [Yang *et al.*, 1992; Foulser *et al.*, 1992; Britanik and Marefat, 1995].
- *Determine process details.* To select cutting parameters such as those shown in Figure 8(c), IMACS uses the recommendations of the Machinability Data Center's handbook [Machinability Data Center, 1980]. The maximum recommended cutting parameters are used, rather than attempting to select optimal cutting parameters; thus IMACS's estimates involve considerable approximation.

As shown in Figure 9, these steps correspond to a task decomposition somewhat analogous to that used in HTN planning [Sacerdoti, 1977; Tate, 1977; Wilkins, 1990; 1988; Yang, 1990; Kambhampati and Hendler, 1992; Erol *et al.*, 1995a; 1994].

Since each FBM can lead to several different operation plans, IMACS does the above steps inside a depth-first branch-and-bound search, evaluating the plans as described in Section 2.6 in order to find the optimal operation plan. For example, Figure 8 shows the operation plan IMACS finds for the socket P_0.

Table 1: Estimated production time for the operation plan shown in Figure 8.

Operation	Time (min)	Operation	Time (min)
drill h1	2.3	mill s2	5.0
drill h3	0.3	mill s4	5.0
drill h5	0.3	mill s6	5.0
drill h7	0.6	mill s8	5.0
drill h9	0.6	mill s9	4.0
drill h11	0.3	mill s10	4.2
drill h12	0.3	3 setups	6.0

Total Time: 39 minutes

2.6 Operation Plan Evaluation

Once IMACS has found an operation plan, it evaluates whether the plan can achieve the design tolerances. To verify whether a given operation plan will satisfy the design tolerances, IMACS must estimate what tolerances the operations can achieve. Typical approaches for computer-aided tolerance charting are computationally very intensive, and only consider limited types of tolerances [Ji, 1993; Mittal *et al.*, 1990]. Thus, IMACS simply evaluates the manufacturability aspects of a wide variety of tolerances without getting into optimization aspects, as described in [S. Gupta and Nau, 1995]. As an example, the operation plan shown in Figure 8 satisfies the tolerances shown in Figure 3, and thus is an acceptable way to make P_0 from S_0.

If the plan can achieve the design tolerances, then IMACS estimates the plan's manufacturing time. The total time of a machining operation consists of the cutting time (when the tool is actually engaged in machining), plus the non-cutting time (tool-change time, setup time, etc.). Methods have been developed for estimating the fixed and variable costs of machining operations; our formulas for estimating these costs are based on standard handbooks related to machining economics, such as [Winchell, 1989; Wilson and Harvey, 1963]. As an example, Table 1 shows the estimated production time for the operation plan of Figure 8.

2.7 Efficiency Considerations

As described in [S. Gupta *et al.*, 1994b; S. Gupta and Nau, 1995], IMACS uses a depth-first branch-and-bound search to generate and evaluate FBM's and plans one at a time. By evaluating them as they are being generated and keeping track of the best one it has seen so far, IMACS can discard FBM's and plans that look unpromising, even before they have been fully generated. For example, from the 22 primary features shown in Figure 6 one can form 512 FBM's for the socket P_0, but IMACS generates only 16 of these FBM's. Below are some of IMACS's pruning criteria, which can be thought of as similar to critics in HTN planning:

- IMACS will discard an FBM if it contains features whose dimensions and tolerances appear unreasonable. Examples would include a hole-drilling operation having too large a length-to-diameter ratio;

a recess-boring operation having too large a ratio of outer diameter to inner diameter; two concentric hole-drilling operations with tight concentricity tolerance and opposite approach directions.

- IMACS will discard an FBM if it appears that there will be problems with work-holding during some of the machining operations. Currently, IMACS's work-holding analysis is based on the assumption that a flat-jaw vise is the only available fixturing device [Das *et al.*, 1994; 1995], but we are currently developing some more sophisticated fixturability analysis techniques that allow the use of both vise clamping and toe clamping.
- IMACS will compute a quick lower bound on the machining time required for an FBM or plan, and will discard the FBM or plan if this lower bound is above the time required by the best plan seen so far.

3 Discussion

Since we did not care whether or not we were doing AI planning in IMACS, there are several differences between the techniques used in IMACS and those used in classical AI planning systems. Some of these techniques may be useful for AI planning. For example, IMACS's technique of finding all primary features before beginning to generate plans can be generalized as follows:

- Enumerate the set of all tasks that might ever be relevant. Call this set $\mathcal{F}$.
- Loop:

  - Generate an incomplete plan F as a subset of $\mathcal{F}$
  - If the plan F has a goal interaction that can't be resolved via precedence constraints, discard it. (If a promising plan exists, it will be generated in another loop iteration.)
  - Flesh out the plan (using task decomposition, critics, plan merging, etc.)

This technique should be useful whenever it is feasible to enumerate in advance the set $\mathcal{F}$ of all relevant tasks. More specifically, suppose that we can construct $\mathcal{F}$ in polynomial time, and that each task in $\mathcal{F}$ will need to be achieved at most once. Then every plan we will care to consider is a subset $F \subseteq \mathcal{F}$, and we can generate these plans nondeterministically in polynomial time. If each goal interaction involves at most a constant number of tasks, then we can determine in polynomial time whether whether there are ordering constraints sufficient to make F a successful plan.

This idea helps to explain a puzzling theoretical problem. In the worst case, planning with STRIPS-style operators is PSPACE-complete [Erol *et al.*, 1994; 1995b], but the best known example of STRIPS-style planning is blocks-world planning, which is only NP-complete [N. Gupta and Nau, 1991; 1992]. This discrepancy can be explained by noting that in a blocks-world problem containing n blocks there are only at most $2n$ possible relevant tasks: for each block b, we might want to move b to the table, and if the goal state contains $on(b,c)$ for some c, then we will want to move b to c.

4 Conclusions and Future Work

IMACS shows that it is possible to address manufacturing planning both realistically and in a principled manner. Our work on IMACS has been well accepted by manufacturing researchers, and we have many ideas for further work on IMACS and on other issues relevant to manufacturing.

Furthermore, some of us (together with Jim Hendler at the University of Maryland)[1] are beginning the development of a test bed in which to compare AI and manufacturing techniques. We intend to develop a collection of manufacturing planning problems and solutions (e.g., designs, plans, and planning systems), presented in a way that is accessible to AI planning researchers for use as a test set or benchmark set. We hope that this will help AI researchers discover ways to apply AI techniques to manufacturing planning in a realistic manner, and possibly to discover issues arising in manufacturing that may be useful for AI planning in general.

References

[Britanik and Marefat, 1995] J. Britanik and M. Marefat. Hierarchical plan merging with applications to process planning. In *IJCAI-95*, 1995, to appear.

[Chang and Wysk, 1985] T. C. Chang and R. A. Wysk. *An Introduction to Automated Process Planning Systems*. Prentice-Hall, Englewood Cliffs, NJ, 1985.

[Das *et al.*, 1994] D. Das, S. K. Gupta, and D. Nau. Reducing setup cost by automated generation of redesign suggestions. In K. Ishii, editor, *ASME Computers in Engineering Conference*, pages 159–170, 1994. Best-paper award winner.

[Das *et al.*, 1995] D. Das, S. K. Gupta, and D. Nau. Generating redesign suggestions to reduce setup cost: A step towards automated redesign. *Computer Aided Design*, 1995, to appear.

[Erol *et al.*, 1994] K. Erol, J. Hendler, and D. S. Nau. HTN planning: Complexity and expressivity. In *AAAI-94*, 1994.

[Erol *et al.*, 1995a] K. Erol, J. Hendler, and D. S. Nau. Complexity results for hierarchical task-network planning. *Annals of Mathematics and Artificial Intelligence*, 1995, to appear.

[Erol *et al.*, 1995b] K. Erol, D. S. Nau, and V. S. Subrahmanian. Complexity, decidability and undecidability results for domain-independent planning. *Artificial Intelligence*, 1995, to appear.

[Famili *et al.*, 1992] F. Famili, D. S. Nau, and S. Kim, editors. *Artificial Intelligence Applications in Manufacturing*. AAAI Press/MIT Press, 1992.

[Foulser *et al.*, 1992] D. Foulser, M. Li, and Q. Yang. Theory and algorithms for plan merging. *Artificial Intelligence*, 57(2-3):143–182, 1992.

[1] We are expecting ARPA funding for this project in the near future. The PI's are Jim Hendler (hendler@cs.umd.edu) and Dana Nau (nau@cs.umd.edu). The work will be carried out jointly with Steve Ray (ray@cme.nist.gov) at NIST. We solicit your input!

[N. Gupta and Nau, 1991] N. Gupta and D. S. Nau. Complexity results for blocks-world planning. In *Proc. AAAI-91*, 1991. Honorable mention for the best paper award.

[N. Gupta and Nau, 1992] N. Gupta and D. S. Nau. On the complexity of blocks-world planning. *Artificial Intelligence*, 56(2-3):223–254, Aug. 1992.

[S. Gupta and Nau, 1995] S. K. Gupta and D. S. Nau. A systematic approach for analyzing the manufacturability of machined parts. *Computer Aided Design*, 27(5), 1995, to appear.

[S. Gupta *et al.*, 1994a] S. K. Gupta, T. R. Kramer, D. S. Nau, W. C. Regli, and G. Zhang. Building MRSEV models for CAM applications. *Advances in Engineering Software*, 20(2/3):121–139, 1994.

[S. Gupta *et al.*, 1994b] S. K. Gupta, D. S. Nau, W. C. Regli, and G. Zhang. A methodology for systematic generation and evaluation of alternative operation plans. In [Shah *et al.*, 1994], pages 161–184.

[S. Gupta *et al.*, 1994c] S. K. Gupta, W. C. Regli, and D. S. Nau. Integrating DFM with CAD through design critiquing. *Concurrent Engineering: Research and Applications*, 2(2), 1994.

[S. Gupta *et al.*, 1995] S. Gupta, W. Regli, and D. Nau. Manufacturing feature instances: Which ones to recognize? In *ACM Solid Modeling Conference*, 1995, to appear.

[Ham and Lu, 1988] Inyong Ham and Stephen C.-Y. Lu. Compute-aided process planning: The present and the future. *Annals of the CIRP*, 37(2):591, 1988.

[Hayes, 1995] C. Hayes. Using a manufacturing constriant network to identify cost critical areas of designs. *Artificial Intelligence in Engineering Design and Manufacturing* (special issue on innovative approaches to concurrent engineering), May 1995, to appear.

[Ji, 1993] Ping Ji. A tree approach for tolerance charting. *International Journal of Production Research*, 31(5):1023–1033, 1993.

[Kambhampati and Hendler, 1992] S. Kambhampati and J. Hendler. A validation structure based theory of plan modification and reuse. *Artificial Intelligence*, May 1992.

[Kambhampati *et al.*, 1992] S. Kambhampati, M. Cutkosky, J. Tenenbaum, and S. H. Lee. Integrating general purpose planners and specialized reasoners: Case study of a hybrid planning architecture. *IEEE Trans. on Systems, Man and Cybernetics* (special issue on planning and scheduling), 1992.

[Machinability Data Center, 1980] Machinability Data Center. *Machining Data Handbook*. Metcut Research Associates, Cincinnati, Ohio, third edition, 1980.

[Mittal *et al.*, 1990] R. O. Mittal, S. A. Irani, and E. A. Lehtihet. Tolerance control in the machining of discrete components. *Journal of Manufacturing Systems*, 9(3):233–246, 1990.

[Nau and Chang, 1986] D. S. Nau and T. C. Chang. Hierarchical representation of problem-solving knowledge in a frame-based process planning system. *Jour. Intelligent Systems*, 1(1):29–44, 1986.

[Nau, 1987] D. S. Nau. Automated process planning using hierarchical abstraction. *TI Technical Journal*, pages 39–46, Winter 1987. Award winner, Texas Instruments 1987 Call for Papers on AI for Industrial Automation.

[Nevins and Whitney, 1989] J. L. Nevins and D. E. Whitney, editors. *Concurrent Design of Products & Processes*. McGraw-Hill, 1989.

[Opas and Mäntylä, 1994] Jussi Opas and Martti Mäntylä. Feature-based part programming. In [Shah *et al.*, 1994], pages 239–260.

[Regli *et al.*, 1994] W. C. Regli, S. K. Gupta, and D. S. Nau. Feature recognition for manufacturability analysis. In K. Ishii, editor, *ASME Computers in Engineering Conference*, pages 93–104, 1994.

[Regli *et al.*, 1995] W. C. Regli, S. K. Gupta, and D. S. Nau. Extracting alternative machining features: An algorithmic approach. *Research in Engineering Design*, 1995, to appear.

[Sacerdoti, 1977] E. D. Sacerdoti. *A Structure for Plans and Behavior*. American Elsevier, 1977.

[Shah *et al.*, 1994] J. Shah, M. Mäntylä, and D. S. Nau, editors. *Advances in Feature Based Manufacturing*. Elsevier/North Holland, 1994.

[Tate, 1977] A. Tate. Generating project networks. In *Proc. IJCAI-77*, 1977.

[Vandenbrande and Requicha, 1993] J. H. Vandenbrande and A. A. G. Requicha. Spatial reasoning for the automatic recognition of machinable features in solid models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(12):1269, Dec. 1993.

[Wilkins, 1988] D. E. Wilkins. *Practical Planning: Extending the Classical AI Planning Paradigm*. Morgan Kaufmann, San Mateo, CA, 1988.

[Wilkins, 1990] D. E. Wilkins. Domain-independent planning: Representation and plan generation. In J. Allen, J. Hendler, and A. Tate, editors, *Readings in Planning*, pages 319–335. Morgan Kaufmann, 1990. Originally appeared in *Artificial Intelligence* 22(3), April 1984.

[Wilson and Harvey, 1963] F. W. Wilson and P. D. Harvey. *Manufacturing Planning and Estimating Handbook*. McGraw Hill, 1963.

[Winchell, 1989] W. Winchell. *Realistic Cost Estimating for Manufacturing*. Society of Manufacturing Engineers, 1989.

[Yang *et al.*, 1992] Q. Yang, D. S. Nau, and J. Hendler. Merging separately generated plans with restricted interactions. *Computational Intelligence*, 8(2):648–676, Feb. 1992.

[Yang, 1990] Q. Yang. Formalizing planning knowledge for hierarchical planning. *Computational Intelligence*, 6:12–24, 1990.