DESIGN FOR MANUFACTURABILITY METHODOLOGY AND DATA REPRESENTATION FRAMEWORK FOR MACHINED COMPONENTS

by

Krishna Kumar Krishnan

Dissertation submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree

of

DOCTOR OF PHILOSOPHY

in

INDUSTRIAL AND SYSTEMS ENGINEERING

Approved:

Osama Eyada

Dr. O.K. Eyada, Chairman

Dr. M.P. Deisenroth

W. J. Fabrycky

Dr. W.J. Fabrycky

Dr. R.D. Dryden

Dr. A. Myklebust

December 1994

Blacksburg, Virginia

DESIGN FOR MANUFACTURABILITY METHODOLOGY AND DATA REPRESENTATION FRAMEWORK FOR MACHINED COMPONENTS

by

Krishna Kumar Krishnan

Committee Chairman: Osama K. Eyada
Industrial and Systems Engineering

(ABSTRACT)

The traditional product development process has been sequential in nature, with the product going through design, process planning, manufacturing and assembly. This sequential decision making results in increased costs and higher product development times. With the trend towards better product quality, product customization, shorter product life cycle, and international competition, manufacturers are faced with the challenge of improving product quality while reducing product development time, manufacturing lead-time, and product cost. To cope with these challenges, the product development process has to be made more efficient by integrating manufacturing and assembly considerations in the design phase itself, through the use of techniques such as Design For Manufacturability (DFM) and Design For Assembly (DFA).

DFM techniques have to be automated to take advantage of the vast advances in CAD and CAM systems. However, the automation of DFM has been constrained, especially for machined components, by the lack of methodologies which are dependent on the process of manufacture, and the incomplete part data representation in CAD systems. This research created a DFM methodology for machined components, along with an appropriate data representation scheme. Also, a software prototype was developed to

demonstrate and validate both the methodology and the data structure.

The DFM methodology consists of three modules: DFM feasibility, process plan generation, and DFM analysis. The DFM feasibility module performs an initial feasibility check on the material, dimensions, tolerances, and configuration of the part. It also generates the spatial relationships between features. The process plan generation module uses a sequence identifier algorithm to generate the manufacturing sequence. The DFM analysis module evaluates tolerances relative to their stacking effects and manufacturability. It then analyzes the part configuration for possible design and process plan improvements.

A software prototype was developed using C++. It addresses the dimension checking, tolerance checking, configuration checking and spatial relationships generation in the DFM feasibility module. In the process plan generation module, the sequence of surfaces/features to be generated has been automated. This sequence is one of the major inputs to a computer-aided process planning module.

Other methodologies for non-machined components can be easily integrated into the DFM framework for complete automation of DFM analysis.

This body of work is dedicated to
my late mother

ACKNOWLEDGMENTS

I would like to express sincere thanks and deep gratitude to my advisor and chairman, Dr. Osama K. Eyada, for his endless patience and encouragement.

Thanks are also due to Dr. M. Deisenroth, Dr. R. Dryden, Dr. W. Fabrycky, and Dr. A. Mykelbust for serving on my committee.

My thanks also go to the following people for their friendship and support during my years at Virginia Tech: Joni, Keith, Larry, Kelcie, Randy, Jeff, Lovedia, Jin, Charles and Andre.

Most of all I would like to thank my wife and daughter for their patience and understanding.

TABLE OF CONTENTS

Abstract ii

Acknowledgments v

List of Illustrations viii

List of Tables x

List of Acronyms xi

Chapter 1: Introduction 1

- 1.1 Design For Manufacturability 3
- 1.2 DFM Automation 5
- 1.3 Problem Definition 7
- 1.4 Research Objectives 14
- 1.5 Dissertation Outline 14

Chapter 2: Literature Review 15

- 2.1. Design For Manufacturability 16
  - 2.1.1 DFM Methodologies 16
  - 2.1.2 Product-Based DFM Work 19
  - 2.1.3 Process-Based DFM Work 21
- 2.2 Data Representation Schemes 24
  - 2.2.1 CAD Data Representation Schemes 24
  - 2.2.2 Feature Recognition Techniques 31
  - 2.2.3 Feature Based Design 41
  - 2.2.4 Product Data Exchange Standard 43
  - 2.2.5 Spatial Relationship Between Features 44
- 2.3 Summary 45

Chapter 3: Research Methodology 47

- 3.1 DFM Methodology For Machined Components 47
  - 3.1.1 DFM Feasibility Module 49
  - 3.1.2 Process Plan Generation Module 57

3.1.3 DFM Analysis Module 59

3.2 Data Representation Scheme 61

3.2.1 Data Structure 65

3.2.2 Feature Class 68

3.2.3 Geometry Class 70

3.2.4 Topology Class 72

3.2.5 Dimension Class 72

3.2.6 Tolerance Class 72

3.2.7 Spatial Relationships Class 75

3.2.8 Process Plan Class 78

Chapter 4: Case Studies on DFM Methodology 81

4.1 Case Study 1 - Drill Sleeve 81

4.1.1 DFM Feasibility Analysis for Drill Slleve 81

4.1.2 Process Plan Sequence for Drill Sleeve 85

4.1.3 DFM Analysis for Drill Sleeve 89

4.2 Case Study 2 - V-Block 91

4.2.1 DFM Feasibility Analysis for V-Block 94

4.2.2 Process Plan Sequence for V-Block 95

4.2.3 DFM Analysis for V-Block 99

Chapter 5: Software Prototype 101

5.1 Software Prototype 101

5.2 Example 1 - E-Shaped Part 105

5.3 Example 2 - Round Part 110

5.4 Example 3 - V-Block 115

Chapter 6: Conclusions and Future Work 119

References 123

Bibliography 131

Vita 144

List of Illustrations

Figure 1.1 Traditional Product Development Process 2
Figure 1.2. An Ideal DFM Framework 8
Figure 1.3 Need For Adjacency Relationships Between Features 10
Figure 1.4 Need For Spatial Relationships Between Features 11
Figure 1.5 Need to Retain Topology of a Feature 13

Figure 2.1 DFM Framework for Welding 17
Figure 2.2 DFM Framework for Sheet Metal 18
Figure 2.3 Extraction of Manufacturing Feature 25
Figure 2.4 Constructive Solid Geometry Model 28
Figure 2.5 Alternative Representation of CSG Model 29
Figure 2.6 B-rep Model and its Data Representation 30
Figure 2.7 Pattern Recognition Technique 32
Figure 2.8 A Geometric Model and its Edge-Face Graph 35
Figure 2.9 Feature Definition Varies 42

Figure 3.1 Traditional Design Process 48
Figure 3.2 Integration of DFM into Design 50
Figure 3.3 Detailed DFM Module 51
Figure 3.4 DFM Feasibility and Analysis Module 52
Figure 3.5 DFM Feasibility Module 53
Figure 3.6 Dimension from Two Sources 55
Figure 3.7 Tool Accessibility Change 56
Figure 3.8 DFM Analysis Module 60
Figure 3.9 Improvement in Design 62
Figure 3.10 Improvement in Process Plan 63
Figure 3.11 Data Representation Framework 66
Figure 3.12 Data Structure 67

Figure 3.13 Description of a Thru_slot 69
Figure 3.14 Example for Partial Accessibility 71
Figure 3.15 Dimension Class 73
Figure 3.16 Tolerance Class 74
Figure 3.17 Specification of Spatial Relationships 76
Figure 3.18 Specification of Process Plan 79

Figure 4.1 Initial Drill Sleeve Design with Dimensions 82
Figure 4.2 Initial Drill Sleeve Design with Labels 83
Figure 4.3 Drill Sleeve Design After DFM Feasibility Analysis 86
Figure 4.4 Initial V-Block Design with Dimensions 92
Figure 4.5 Initial V-Block Design with Labels 93
Figure 4.6 V-Block Design After DFM Feasibility Analysis 96

Figure 5.1 System Architecture 103
Figure 5.2 User Menu System 106
Figure 5.3 Software Flow Diagram 107
Figure 5.4 Dimensioned E-Shaped Part 108
Figure 5.5 Labeled E-Shaped Part 109
Figure 5.6 Dimensioned Round Part 112
Figure 5.7 Labeled Round Part 113
Figure 5.8 Dimensioned V-Block 116
Figure 5.9 Labeled V-Block 117

List of Tables

Table 3.1 Data Requirements 64

Table 4.1 Dependency Matrix for Drill Sleeve 87
Table 4.2 Process Sequence Matrix for Drill Sleeve 88
Table 4.3 Alternate Process Sequence Matrix for Drill Sleeve 90
Table 4.4 Dependency Matrix for V-Block 97
Table 4.5 Process Sequence Matrix for V-Block 98
Table 4.6 Alternate Process Sequence Matrix for V-Block 100

Table 5.1 Software Prototype Specifications 102
Table 5.2 DFM Feasibility Analysis Output for E-Shaped Part 111
Table 5.3 Process Sequence for Round Part 114
Table 5.4 Process Sequence for V-Block 118

List of Acronyms

AI	Artificial Intelligence
B-rep	Boundary Representation
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CAPP	Computer Aided Process Planning
CIM	Computer Integrated Manufacturing
CNC	Computer Numerical Control
CSG	Constructive Solid Geometry
DFM	Design For Manufacturability
DFA	Design For Assembly
DFI	Design For Inspection
IGES	Initial Graphics Exchange Standard
OOP	Object Oriented Programming
PDES	Product Data Exchange Standard

CHAPTER 1

INTRODUCTION

The traditional product development process is of the sequential type in which the product goes through design, process planning, manufacturing, and sales (see Figure 1.1). This sequential process of decision making results in increased costs through the duplication of engineering effort, mostly through redesign [Stol89]. Although the design cost of a product is approximately 5% of the total product cost, the design usually determines more than 70% of the manufacturing costs [Hoth90, Bloo89]. It also influences 60% of the product quality and 50% of the manufacturing lead time [Bloo89].

With the trend towards better product quality, product customization, shorter product life cycle, and international competition, manufacturers are faced with the challenge of improving product quality while reducing product development time, manufacturing lead time, and product cost. This challenge is magnified by the rapid development of newer manufacturing technologies, including automation. Cost reductions of fifty percent or more are needed to remain competitive [Hoth90, Bloo89]. Cost reductions of this magnitude can be only achieved through the integration of the various product life-cycle aspects early into the design phase. Product development methodologies, such as concurrent/simultaneous engineering, design for manufacturability (DFM), and design for assembly (DFA), aim at facilitating these integration requirements.

Concurrent engineering has been defined as:

> "a systematic approach to the integrated, concurrent design of products and their related processes including manufacture and support. This approach is intended to cause the developers, from the outset, to consider all elements of the product life cycle from conception through

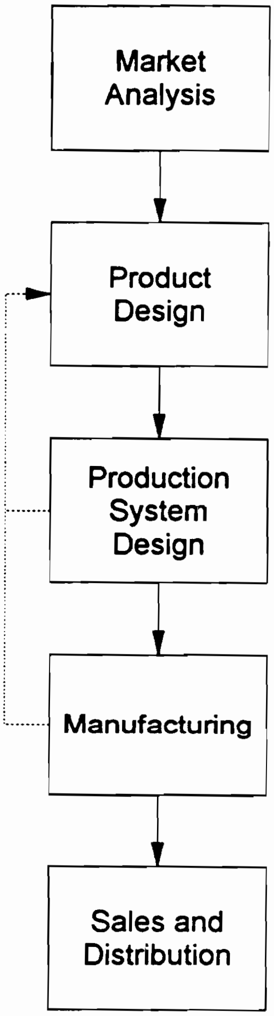


Figure 1.1 Traditional Product Development Process [Stol89]

disposal, including quality, cost, schedule, and user requirements" [Winn88].

Concurrent or simultaneous engineering methodologies depend upon forming multi-disciplinary design teams consisting of personnel from design, manufacturing, sales, quality control and other relevant sectors to assist and provide information to the designer. Various product development aspects are considered simultaneously and this often increases creativity by stimulating discussion on potential trade-offs.

On the other hand, DFM and DFA are limited in that they focus on manufacturing and/or assembly only. DFM involves the incorporation of manufacturing considerations into design to make the product amenable to manufacture, while DFA focuses on the integration of assembly requirements. This research focuses on DFM as applied to machining components.

1.1 Design For Manufacturability

Examination of the literature indicates that a commonly accepted definition of DFM does not exist. Some of the definitions used are given below:

> "a product development process that is characterized by the cooperative participation of members of the management, research, marketing, design, production, distribution, and support divisions throughout a products life" [Bloo89]
>
> "sever the dichotomy of design and manufacturing practices so that these two areas will be pursued as a homogenous activity" [Dasi90]
>
> "the integration of product design and process planning into one common activity" [Li88]
>
> "understanding how product design interacts with the other

components of the manufacturing system and in defining product design alternatives which help facilitate 'global' optimization of the manufacturing system as a whole. Design for Fabrication (DFF), involves the design of product components and parts in ways which are compatible with the method of manufacturing" [Stol91]

"DFM is the inclusion of the value engineering process in the design phase itself. Traditionally the value engineering department is placed outside of the design to manufacturing phase." [Corb91]

"a product development practice which incorporates manufacturability concerns into the design process" [Rose92a]

From the above it can be concluded that DFM is a value analysis of the part design with respect to manufacture. In the traditional product development process, this is normally performed during process planning. The value analysis is composed of dimensional, tolerance, material and part configuration analysis. However, in DFM, these value analyses should be integrated into the design process. Dimensional analysis involves checking the accuracy and completeness of part dimensions and for proper constraining of the model. In tolerance analysis, the part design has to be checked for appropriate tolerances, the ability of available manufacturing resources to attain them, etc. The material analysis consists of checking the compatibility of the material with the process of manufacture (machinability, castability, etc.) and to available tooling, replacement with a less expensive material, etc. The compatibility of the geometry of the part with the process selected is determined during part configuration analysis, such as tool accessibility for metal cutting process and material flow for casting processes. The impact of the process plan may also be analyzed to determine tolerance stacking and improvement of process plan. These value analyses have to be performed with respect to the process to be used in manufacturing the part.

1.2 DFM Automation

With the exception of very few software packages with limited scope and objective, concurrent engineering as well as DFM and DFA are manually applied through committees. This is expensive, time consuming, and prone to error.

Automation of concurrent engineering, DFM and DFA can prove to be an effective tool in the integrated product development process. The advent of computers has impacted all phases of the product development process, both in terms of the speed of product development and with respect to the intricacies of the products being developed. An integrated product development process requires automation of all aspects of the product development process, including DFM. An automated DFM system will serve as a bridge between CAD and CAM systems, which are still isolated and exist as "islands of automation."

Automation of DFM has been constrained by two major factors:

- Lack of methodologies to be used in the DFM analysis
- Lack of information required for manufacturing analysis in CAD data representation

Before a DFM analysis can be automated, it is important to have an appropriate methodology. Examination of the literature indicates two methodologies. For sheet metal parts, Yu et al. [Yu92] developed a method for the manufacturing assessment of a 3D part followed by unfolding it to assess its flat pattern. For large welded structures, Eyada et al. [Eyad93a] described a method for the integration of arc welding parameters into design to optimize manufacturing costs. The above methodologies are heavily dependent on the process selected, because of the variation in the data and knowledge required for the DFM analysis. Therefore, it is necessary that the methodology should conform to the characteristics or needs of the process of manufacture. Also, it should be designed to make the data and manufacturing knowledge necessary for DFM analysis part of the design process.

For the effective integration of the product development process a common part data representation which is amenable to both CAD and CAM (DFM, DFA, and process planning) has to be developed. The explicit representation of manufacturing information is necessary for the automation and effective DFM analysis of the part. Manufacturing processes require part information in terms of features and the relationships between them, while CAD systems store data in the form of entities such as vertices, edges, and surfaces and the adjacency relationships between them.

The higher-level feature information required by CAM systems can be generated either through the use of human assisted feature recognition, automatic feature recognition and extraction techniques (using algorithms) or through the use of feature based design systems [Shah88]. Human assisted feature recognition involves the identification of features by an user-interactive selection of entities from the CAD display screen [Chan84]. Feature recognition consists of building up the geometric and topological relationships of the model from a CAD system, analyzing these relationships, and isolating those areas that conform to a manufacturing feature [Eyad93b]. Feature based design systems use features as primitives in the modeling of the part [Chan89, Chen91b, Chen92, Dixo88a, Gand89, Li91, Lin92, Padh90, Shah88].

The data necessary for DFM analysis depends on the process selected, adding to the complexity of the problem. For example if the process selected is casting, the data necessary might be the features and the location of gates, risers, spruces, etc. While, if the process selected is machining, the data necessary are material-tool compatibility, tool accessibility, sequence of surfaces to be machined, etc. The manufacturing knowledge necessary to perform the analysis is also dependent on the process selected.

In summary, an ideal DFM framework should take into consideration process specific information (Figure 1.2). The methodology for each process has to be developed before DFM analysis can be automated. The process specific information has to be generated after the process is selected and the appropriate knowledge base has to be attached to the appropriate DFM modules.

1.3 Problem Definition

Literature survey of DFM principles and methodologies show that efforts have mostly been directed towards net-shaped manufacturing processes such as, casting, molding, etc. [Iran89, Ishi90, Ishi92a, Ishi92b, Lu88, Padh90, Rose92a]. Other works have focused on sheet metal stamping [Yu92 & Schm89a], machining [Cutk89, Dasi90, Laak90], CNC manufacture [Hodg91] and welding [Eyad93a].

For machining components, Cutkosky et al. [Cutk89] proposed the simultaneous design of the part and the process plan, which requires a great deal of manufacturing knowledge on the part of the designer. This also requires the designer to decide upon the process of manufacture prior to the design of the part. Da Silva et al. [Dasi90] pointed out the importance of tool accessibility and provided few rules to determine tool accessibility. However, due to the lack of explicit representation of tool accessibility information, additional computing would be necessary for the automation of these rules. A feature-based CAD system which incorporates manufacturability analysis was described by Laakko et al. [Laak90]. The manufacturability analysis is restricted to individual feature analysis and not with respect to the inter-relationship between the features.

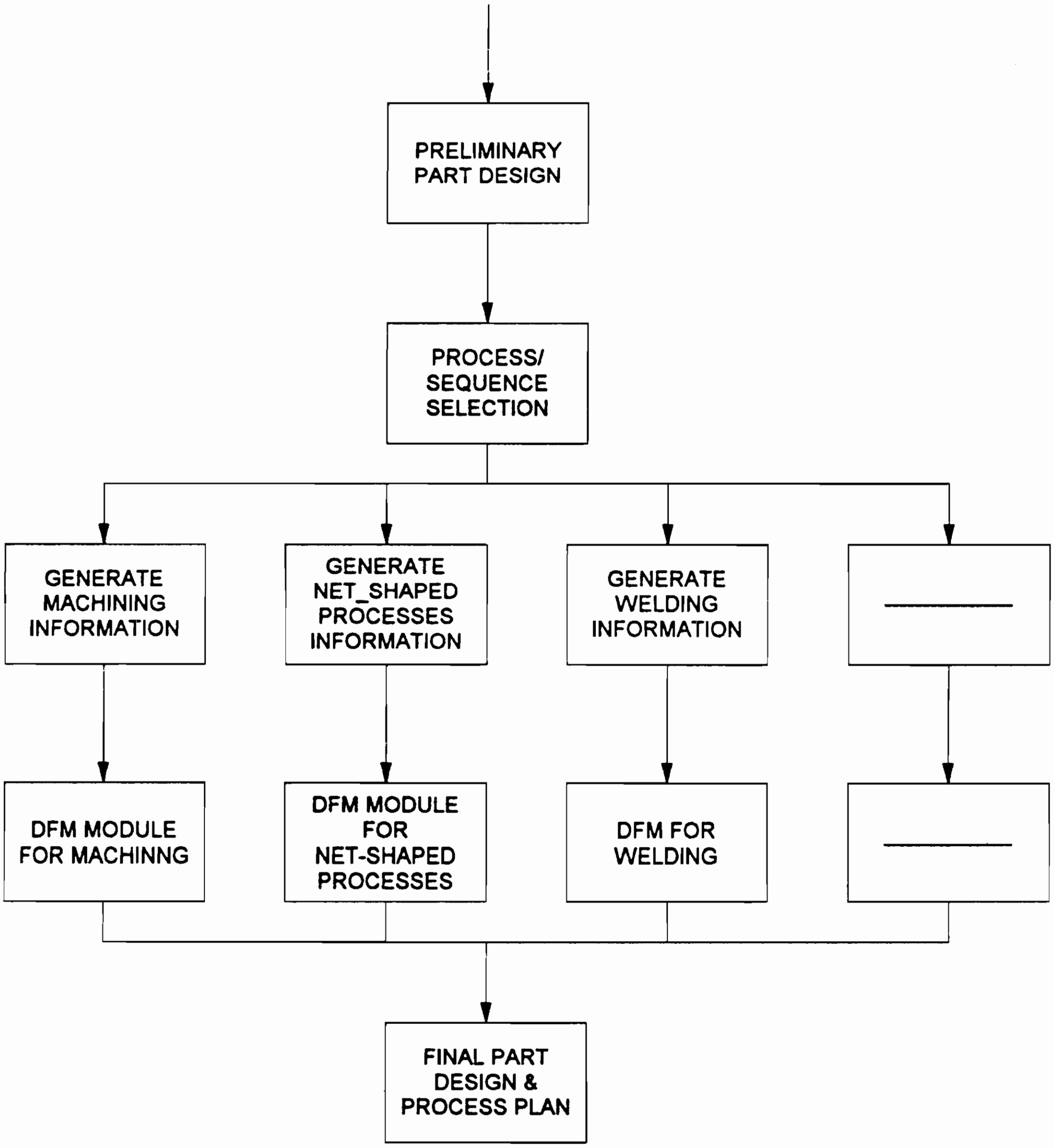


Figure 1.2 An Ideal DFM Framework

Finally, Hodgeson [Hodg91] proposed that for CNC manufacture the part should be broken down to two or more subparts to facilitate a single direction (or setup) of manufacture and assembly. Clearly, this approach will result in increasing the number of components and the design complexity.

From the above, it can be concluded that there has not been any substantial research towards the development of a DFM framework or in the automation of its analysis for machined components. A large number of products are produced using metal removal processes. Machining can be used to generate any regular geometry, with high tolerances. However, the degree of complexity in computerization is much higher than other processes. Process planning is much more complex, with larger amount of data manipulation and with a series of setups.

DFM for metal removal processes pose some unique challenges. Features have been identified as the necessary link for data transfer between CAD and CAM. However, the issue of explicit representation of spatial relationships between features needs to be addressed for the effective automation of the DFM and DFA modules, especially for metal removal processes [Fing89]. For instance, consider the part model in Figure 1.3. The hole if analyzed by itself can be drilled very easily. But it is difficult to manufacture this hole because of the drill drift caused by the slanted surface. The feature by itself is not relevant. What is more important is the relationship between the hole feature and the slanted surface on which the hole is. This example highlights the need for spatial relationship between adjacent features. However, adjacency information is not enough. Consider, the hole to be drilled in Figure 1.4. The hole (hole_A) by itself is locally accessible and with respect to the adjacent features is manufacturable. However, in a global sense the feature is inaccessible. It needs the spatial relationship between the hole_1 and

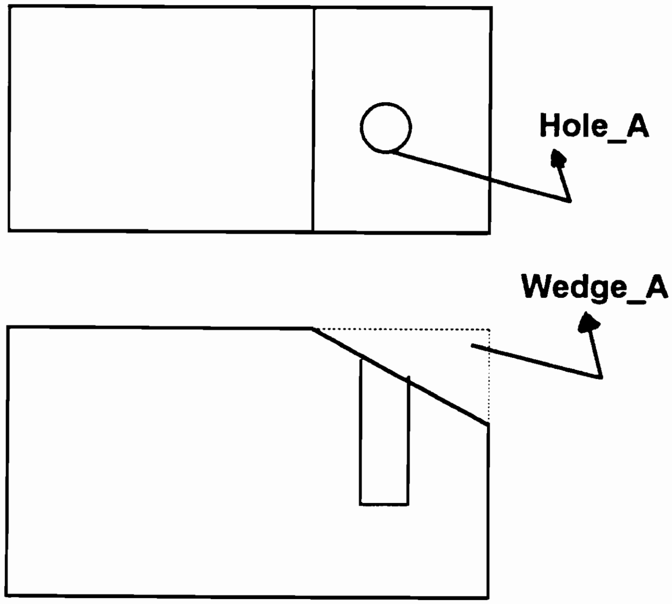


Figure 1.3 Need for Adjacency Relationship between Features

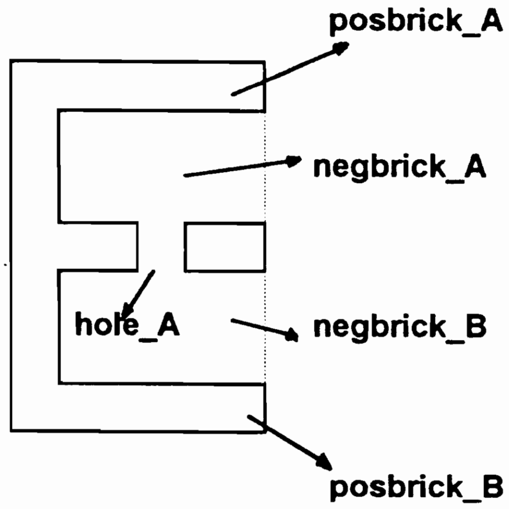


Figure 1.4 Need for Spatial Relationships Between Features

the brick_A to determine the infeasibility. Both the examples highlight the need for spatial relationships between features, especially for machined components.

The process plan also plays a large role in determining whether a product is manufacturable or not. The number of setups used and the sequence of surfaces/features generated can determine whether a part design is economically viable. The process plan and tool accessibility information are critical in determining the manufacturability of the part in terms of the process plan feasibility and part configuration analysis.

In terms of the data required, the topology and geometry undergo changes as the process plan is analyzed. In feature-based design modelers, which do not retain the feature geometry and topology, process planning becomes more cumbersome. Figure 1.5 shows the change in topology due to the addition of a slot on a brick. In process planning, the machining of F_1 is no longer viable, since F_1 has been modified, to be three different faces, F_1, F_7, and F_8. For effective process planning, the data on F_1 should be retained.

In conclusion, an ideal DFM module for machining components should consist of a well defined methodology along with appropriate data representation, especially the explicit representation of the following:

- Features
- Tool accessibility
- Spatial Relationship between features
- Process Plan
- Material
- Dimensions and Tolerances

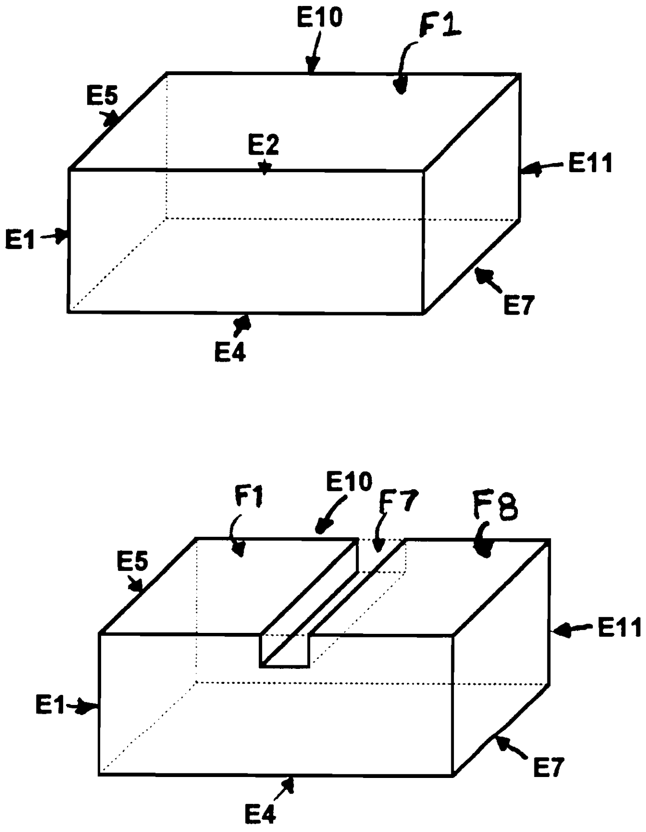


Figure 1.5 Need to Retain Topology of a Feature

1.4 Research Objectives

The primary objective of this research is to create a DFM methodology for metal removal processes. The research also addresses the development of a framework for part data representation as well as the data required for DFM analysis for metal removal processes. A prototype software, developed using C++, with case studies represented in the developed data framework along with information such as tool access directions and spatial relationships, is used to demonstrate the methodology and research problem. The software prototype reads part design information from a file. The research also incorporates some DFM principles in the software to validate the data representation scheme.

The research objectives achieved are as follows:

a) Created a DFM methodology for machining components
b) Validated the methodology with two case studies
c) Determined data necessary for DFM analysis
d) Designed a data representation framework
e) Developed a software prototype for demonstration.

1.5 Dissertation Outline

In Chapter 2, a literature review of all the relevant material is provided. Chapter 3 details the research methodology created for metal removal processes. It also describes the data necessary and the object-oriented data representation framework. In Chapter 4, two case studies are used to demonstrate the manual DFM methodology. Chapter 5 presents the details of the software prototype. Chapter 6 concludes with a discussion on the advantages and limitations of the work presented and provides suggestions for future research.

CHAPTER 2

LITERATURE REVIEW

This chapter provides the literature survey of both design for manufacturability and data representation. The majority of DFM work provides a comparison of the design before and after DFM analysis is applied along with some rational behind product re-design. Only two articles provided either a methodology or a procedure that can be followed to perform some DFM analysis. Also, only two articles are related to metal removal processes. The literature survey was therefore extended to cover directly and indirectly related work on DFM, and is classified based on the type of product or the manufacturing process used. Section 2.1 first reviews DFM methodologies, followed by product and process related DFM research.

For the effective integration of the product development process, a common part data representation which is amenable to both CAD and CAM (DFM, DFA, and process planning) has to be developed. The explicit representation of manufacturing information is necessary for the automation of DFM analysis. Manufacturing processes require part information in terms of features and the relationships between them, while CAD systems store data in the form of entities such as vertices, edges, and surfaces and the adjacency relationships between them. In Section 2.2, the various methods of data representation in CAD systems are first described along with the algorithms developed to extract manufacturing features and feature-based modeling. The form-features-information model in PDES standards are also described. Research work done in the area of representation of spatial relationships between features is then reviewed. Finally, Section 2.3 summarizes the literature survey relative to research needs and its importance.

2.1 Design For Manufacturability

This section reviews the various research efforts in developing DFM methodologies, followed by product related DFM work and finally process related DFM efforts.

2.1.1 DFM Methodologies

Examination of the literature relative to DFM methodologies indicates that there has been no substantial efforts towards the development of DFM methodologies. A systematic, detailed DFM methodology or procedure does not exist. Only two methodologies were found in the literature that are worthy of reporting. Eyada et al. [Eyad93a] proposed a DFM framework for the integration of arc welding factors into design of large welded structures (e.g., ships). Factors affecting the design, manufacture, economic, and quality of large welded structures were first identified, and those with the highest impact on manufacturing costs were integrated into the design process. Figure 2.1 shows the overall structure of the developed DFM framework. The framework is further expanded to provide the details of the implementation. Although this is the most detailed DFM methodology that can be found in the literature, the methodology is limited to arc welding processes as applied to large welded structures. Also, a data representation for the full automation of the methodology is not available.

Yu et al. [Yu92] proposed a DFM framework for sheet metal fabrication of parts made by cutting, bending, flanging, and local stretching only (see Figure 2.2). The system uses a feature-based CAD modeler. Tasks, rules and design guidelines for manufacturing assessment have been developed to assist the designer for both 3D and flat pattern (or unfolded 3D) analysis. The main advantage of this methodology is the capability of 3D assessment and the use of

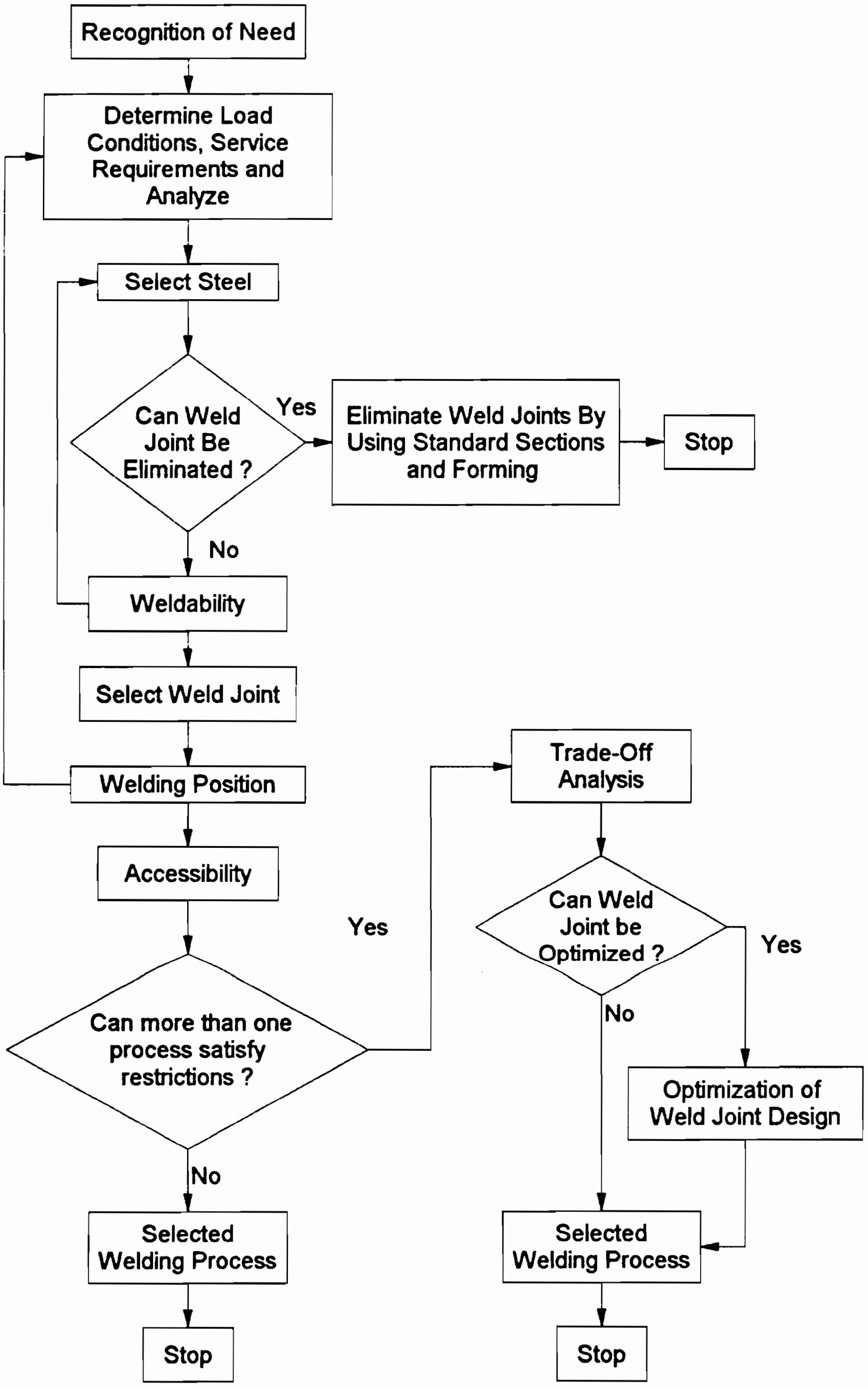


Figure 2.1 DFM Framework For Welding [Eyad93a]

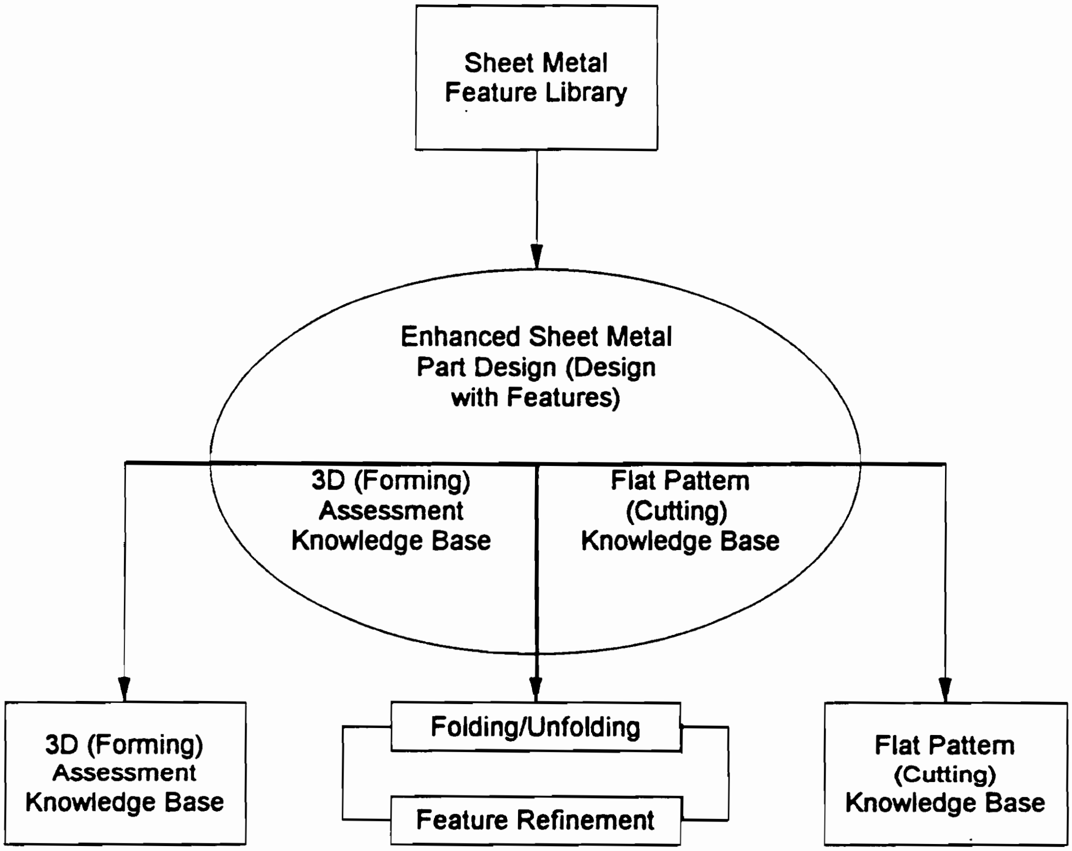


Figure 2.2 DFM Framework For Sheet Metal [Yu92]

feature information including their relationships for DFM analysis. The 3D assessment is based on bending, reinforcement and feature relationships. The grain lamination, bend radius, etc., are used in bending evaluation. Spatial relationships (limited to adjacency relationships) between the features (e.g. bead, rib, and wall) are used for evaluation of the strength of ribs, beads, etc. The flat pattern assessment involves the assessment of the blank used for the sheet metal process in terms of size, dimensions, tolerances, and contours to determine producibility. However, algorithms required for data retrieval and feature refinement are complex, due to the lack of explicit data representation for DFM analysis. This is the most comprehensive methodology which has been automated.

2.1.2 Product-Based DFM Work

Rinderle et al. [Rind89] proposed the use of form-function characteristics to introduce the effect of size and manufacturing variability (tolerances) in the analysis of electro-mechanical designs (i.e., motors). A constraint-based model, which mathematically represents the parameters pertaining to the product functional performance, is used to determine the form-function relationships. Various physical parameters, such as size and tolerances, are optimized for a given functional requirement, such as torque. This work demonstrates the importance of integrating some manufacturing aspects (conventional tolerances) into product design through the derivation of the form-function relationship. This procedure if expanded could provide the designer with a CAD system, which will allow the design to be represented as equations rather than geometry alone. However, the concept of form-function is limited to one type of DFM analysis, conventional tolerances. Material, geometric, and configuration DFM analysis are not addressed by this method, and in turn it can not be extended to other products, such as mechanical products.

Kim et al. [Kim88] proposed a knowledge based manufacturing advisor for the design of turbine disks. The proposed system will allow the user to define the design specifications and analyze the design for optimization and manufacturability. However, the system has not been implemented, its scope is limited to turbine disks, and very little DFM analysis is reported.

Other literature's that are focused on the analysis of products in terms of their manufacturability do not provide enough details on the DFM procedure or the type of trade-offs performed. Deisenroth et al. [Deis92] presented three DFM case studies from some GE plants. The first case study is on the re-design of lighting panel board (circuit-breaker type device). The re-design effort was focused on manufacturing, marketing, finance, etc. and was done through committees. The design was broken down into sub-systems and achieved significant reduction in the number of parts. The second case study involved the re-design of electric motors for golf carts, pumps, etc. to reduce the cost of the current product while building superior characteristics. This effort also resulted in reducing number of parts and an improved motor housing design. However, the new design has not been acceptable to the golf-cart manufacturers due to the loss of sturdiness. The third study involved the design of fluid carrying tubes in aircraft engines using composite material, to take advantage of their higher strength-to-weight ratio. The tubes were originally designed to be made from metal. A re-design of the tubes were performed taking into consideration the characteristics of composite materials. A stand alone software that the desinger can querry regarding the possibility of manufacturing a tube using composite material was developed. The software is very limited in scope, and covers only the braiding process. All the above three case studies, show the before-and-after approach to DFM analysis. They do not provide any procedure for integration of DFM into design.

2.1.3 Process-Based DFM Work

Although most of the DFM publications are based on the type of the process to be used in manufacturing, only net shaped, sheet metal, metal removal, and welding processes have been investigated. These efforts are reviewed below with the exception of the welding related work, which is explained in the DFM methodologies section.

2.1.3.1 Net Shaped Processes

Net shaped processes are those that use molds or dies to transform material which is shapeless, or of simple shape, into a near finished part [Ishi92a].

Chen, Miller and Vemuri [Chen91a, Chen91b, and Chen92] proposed a generic framework for the integration of a design environment with a knowledge based environment for manufacturability assessment of cast products. They use feature based modeling techniques along with an object-oriented part model representation scheme. In addition to feature and geometry information, data stored include adjacency relationships between features and the representation of compound features (combination of two features). Configuration analysis is the only analysis performed. The need for spatial relationships is highlighted. Dimension and tolerance representation issues as well as their analysis are not addressed.

Rosen et al. [Rose92a] developed algorithms for the automatic evaluation of injection molded and die cast components for tooling cost. Tooling cost drivers and its corresponding features for die casting and injection molding, which includes size of component, cavity detail, parting plane complexity, etc. are identified. Algorithms to convert design features to tooling cost features and for the evaluation of the design based on tooling cost are also developed. However, the algorithms can only be used to determine the cost, not to optimize or improve the design.

Ishii and Miller [Ishi92a] analyzed the different methods of design representation for net shaped manufacturing processes. The use of feature-based design is suggested to be not viable due to the large number of features that have to be represented and the associated data that has to be represented to capture the various perspectives (design, tooling, and processing). A compromise between a feature-based design system and a pure geometry oriented system is suggested as the method of design representation. The proposed system will use a limited set of features defined in the feature library and will allow the user to define any additional features required. However, the design rules are available only for the features in the library and require the user defined features to be analyzed by local shape evaluations using the geometrical and topological data available. In addition dimension and tolerance analysis is not addressed.

Irani et al. [Iran89] addressed the automation of gating design for injection mold systems. The inputs to the system are, a feature-based representation of the part, material properties, and machine data, such as maximum injection pressure and maximum clamping force. The system allows two levels of analysis depending on information supplied by the user. In the first level, details of the mold such as gate location, number of gates etc. are user defined. In the second level, the user does not specify any details. At both levels the user is allowed to specify constraints, such as location of gates, vent locations, etc. Trial gating plans are algorithmically generated. The system evaluates the gating plans based on geometry related performance parameters such as venting, jetting, etc., and process related performance parameters such as mold temperature, injection temperature and time etc. The focus of this work is too narrow and requires integration into more detailed casting analysis programs to complete the evaluation.

2.1.3.2 Sheet Metal Processes

Schmitz and Desa [Schm89a and Schm89b] proposed and implemented a DFM analysis for stamped products. They identified manufacturing cost factors, such as type of press, number of punching tools, material scrap, etc. Algorithms to associate part design specifications to manufacturing cost factors and hence the manufacturing costs were generated. Once the manufacturing cost factors are identified, the design specifications which increase the manufacturing cost values are identified and provided to the designer. The designer is allowed to determine the value of the functionality of the design specification and modify it. This system is limited to configuration analysis.

2.1.3.3 Metal Removal Processes

Cutkosky and Tenenbaum [Cutk90] proposed a methodology based on the premise that DFM can be best achieved by simultaneously designing the part and the process used to make it. The part is designed by specifying high level process planning operations, such as sweep, pocket and hole. The process plan can then be developed by breaking down the high level operations into a low level plan. This procedure constrains the designer severely. The designer is forced to think about the design in terms of manufacturing processes, which limits creativity and thinking capabilities, since the designer is more oriented towards thinking in terms of design features.

Hodgson and Pitts [Hodg91] proposed a few design rules to be considered in CNC manufacture. The primary rule that they propose is that of a single-setup manufacture. They argue that components should be broken down to facilitate manufacturing in a single setup. This makes the components easy to inspect, resulting in less complex inspection setups. This would also facilitate a unidirectional assembly sequence, making the parts easy to assemble. However, the methodology is not amenable to automation. It increases the number of components and the design complexity. It also increases the time required for assembly.

2.2 Data Representation Schemes

The importance of the explicit representation of manufacturing information such as feature and the spatial relationships between them for manufacturing activities such as process planning and DFM analysis has been highlighted earlier. CAD part data representation schemes store only information about the entities such as vertices, edges, and surfaces. The higher level feature information required for manufacturing analysis can be obtained either by the use of feature extraction and feature recognition algorithms, which are CAD modeler dependent, or through the use of feature-based modelers (see Figure 2.3). This section provides a brief review of the various CAD data representation schemes. A review of the feature recognition algorithms as well as a brief description of feature based modeling is provided. Finally, work in the area of definition of spatial relationships is reviewed.

2.2.1 CAD Data Representation Schemes

This section briefly describes the most common CAD data representation schemes: wire-frame, surface representation, and solid models.

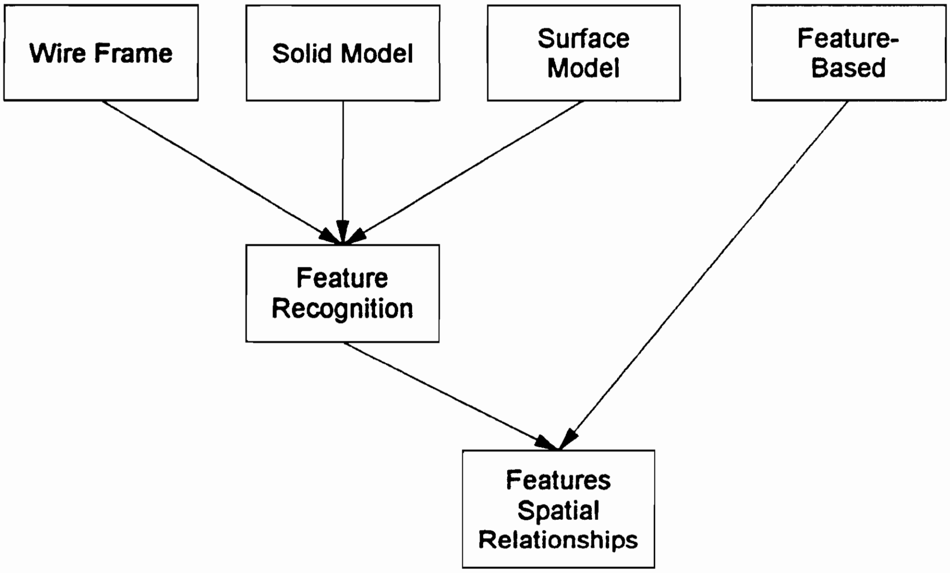


Figure 2.3 Extraction of Manufacturing Features

The wire-frame technique used to be the most widely used method for CAD representation because of its simplicity in development and usage. Under this technique, the object is represented in terms of edges, vertices and curves. This approach suffers from the problem of creating ambiguous objects. Visually a wire-frame object is confusing to the human eye. Additionally, complex objects cannot be very easily modeled. Information on the relationships between the basic geometric entities is not maintained, and in turn obtaining surface information requires some processing. It is possible to construct the topology for wire frame models, and thus most of the aforementioned problems could be resolved. This would combine the advantage of ease of development and use of the CAD system with the advantage of accessing topological information for computer-aided processing planning.

Surface representation models are created by connecting different types of geometric elements of a surface to line segments or boundaries. The advantage of this approach over wire-frame modeling is the availability of surface information. This helps in hidden-surface removal and overcomes the ambiguities of wire-frame models. Also, because of the availability of the structure boundaries, surface representation can be used for NC programming. However, a great deal of processing is still required for building topological information.

Solid modeling techniques are better than the above methods in terms of the level of geometrical and topological relationships provided by its data structure. Solid modeling allows the user to think at a higher level of geometric abstraction. The most popular solid-modeling techniques are Constructive Solid-Geometry (CSG) and Boundary-Representation (B-rep). In CSG technique, the solid model is constructed by the combination of different solid primitives, such as cuboids, cylinders, spheres, and torus. The primitives are combined by the application of Boolean operations including union, difference and intersection. Therefore, the feature information is explicit but the geometric model is implicit. The user can change the size of the primitives and their location and orientation in space. The model can be represented by a binary tree structure (Figure 2.4) in which the terminal nodes are the primitives or a combination of primitives and the non-terminal nodes are the Boolean operators.

A main drawback of CSG modeling is its non-uniqueness. Figure 2.5 shows an alternative binary tree representation of the model in Figure 2.4. This creates a problem for feature recognition since model data can be different for the same object. Lee and Fu [Lee87] proposed a two-step procedure, extraction and unification, to address this problem. In the extraction step, the CSG tree is traversed and all principal axes of the primitives are extracted. Next, the principal axes are clustered together based on certain pre-conditions for feature definitions. In the unification step, the features are first unified and then the CSG model is reconstructed. The end result is a unique representation of the features within the CSG tree. However, this approach still falls short in providing all the information needed for feature recognition, geometric data and topological relationships. This can only be achieved if the boundary representation of the product is available.

In boundary representation (B-rep), the model is built in terms of its faces, edges and vertices. Topological relationships, adjacency relationships between the geometric primitives, as well as surface information are part of the data structure. A B-rep model along with its data structure is shown in Figure 2.6. The feature information in B-Rep is implicit and the geometric model is explicit, which is the reverse of CSG representation. In addition, the adjacency relationships between the geometric primitives must be evaluated for feature recognition purposes.

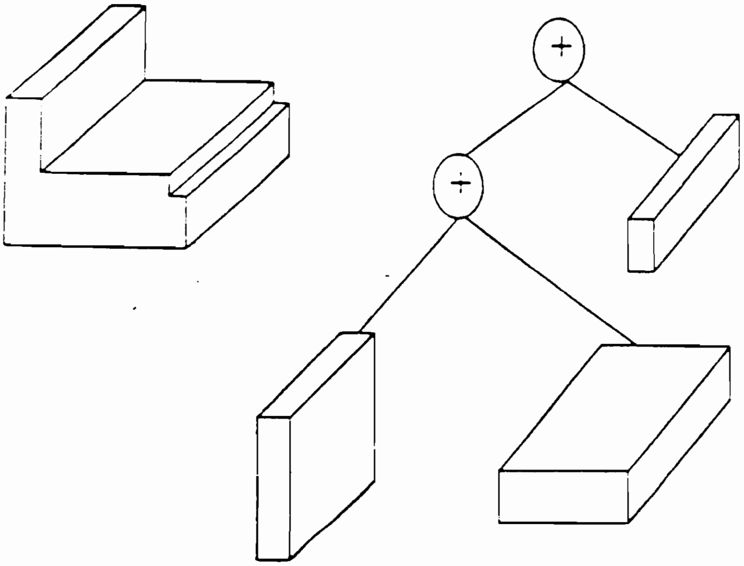


Figure 2.4 Constructive Solid Geometry Model

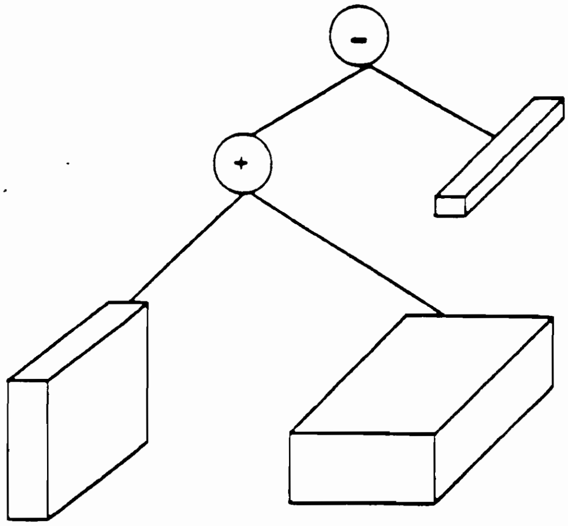


Figure 2.5 Alternate Representation of CSG Model

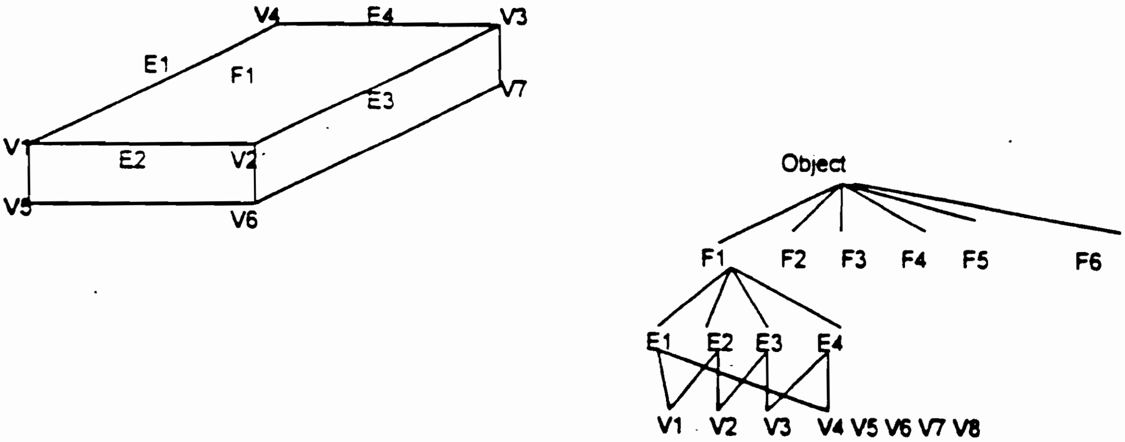


Figure 2.6 B-rep Model and its Data Representation

2.2.2 Feature Recognition Techniques

This section discusses the various techniques, pattern recognition, graph theory, and expert system, used in the feature recognition process. It also reviews some of the research efforts using these techniques.

2.2.2.1 Pattern Recognition Approach

Syntactic pattern recognition systems consist of three major elements: pattern description or representation, structural rules or grammars for combining primitives to describe more objects or classes and syntax analysis (parsing) [Fu82]. During pattern recognition, the geometric model is decomposed into pattern primitives. A parser is used to check the pattern primitives to determine if a set of these primitives can form a pattern class. If a set of pattern primitives match a pattern class, a feature is recognized. Typically, pattern primitives are symbolically defined for each type of basic geometric entity. During the decomposition of a geometric model into pattern primitives, a corresponding pattern primitive is assigned and concatenated into a string for each of the geometric entities encountered. In the example given in Figure 2.7, pattern primitives A, B, and C are defined for the corresponding geometric entities. Pattern primitives can have associated lengths. The string is constructed by analyzing the geometric model, say, from the left to the right. The string formed out of this example is thus A(a)B(b)A(c)C(d)A(e), where the symbols in capitals refer to primitives and the lower case letters in brackets refer to the lengths of the primitive. This type of a string represents a slot and thus a feature is recognized.

Staley, Henderson and Anderson [Stal83] described a system for recognizing holes from a 3-D solid geometric data base. The system requires a string representation of the hole cross section as input. This is achieved by

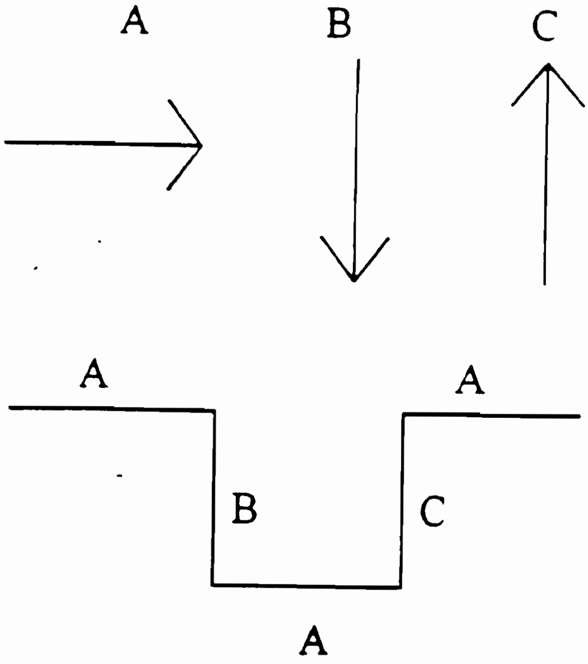


Figure 2.7 Pattern Recognition Technique

having the user interactively select a hole and specify the start and end section vertex on the cross section face. The sectioned object data structure is sent to a Pascal string generator program which traverses the data to generate a string of basic geometric primitives. The string is a 2-D representation of the 3-D data. Only straight line segments are allowed to be used as edges. Once the string representations are created, a lisp recognizer, based on Earley's Parsing Algorithm, is used to check if the input string is a part of the pre-defined hole grammars. Checking is done from the simplest hole pattern to the most complex one. If the input string matches any of the grammars, the system returns a value 'true' and specifies the type of hole that is recognized. The 3-D solid objects were created using the solid modeler Romulus. The system reduces the 3D data to 2D limiting the analysis. The system is also restricted when two or more features intersect, modifying the interacting features. This system was implemented in Lisp on a VAX 11/780 with a UNIX operating system.

Li and Bedworth [Li88] developed a syntactic pattern recognition algorithm to detect rotational features from an Initial Graphics Exchange Specifications (IGES) file for CAPP purposes. Since the input to the algorithm is an IGES file, the algorithm is independent of the CAD modeler used. The data is translated to a 2D representation if necessary and a pattern string associated with the part geometry data is created. The feature recognition algorithm acts on this pattern string, matching the string to predefined features. The first three characters of the pattern string is matched against predefined grammars. If no features are found the first character is dropped and the next rightmost character is picked to form the new set of three characters. The search is continued until the rightmost character is evaluated. A serious limitation of this algorithm is that the parts have to be symmetrical along the X-axis. the algorithms are limited to 2D analysis. The implementation was done using the CAD software CADAM.

2.2.2.2 Graph Theory Approach

Topological relationships between geometric entities of a CAD model can be represented using graphs which simplify the analysis of the geometric entities for feature recognition. Simple heuristics can be applied to the graph to identify features. The topological model can be structured as edge-vertex graph, edge-face graph, etc. In an edge-vertex graph, the nodes represent the vertices of the geometric model and the arcs represent its edges. The adjacency of vertices in the model is mapped accordingly. In an edge-face graph, the nodes represent the faces of the geometric model and the arcs represent its edges. Adjacent nodes represent adjacent faces with the arcs denoting the common geometric edge between them. Figure 2.8 shows an object and a partial face-edge graph. F6 represents a cut node in the graph, i.e. the removal of this node separates the graph into two components, each component representing a feature. The separated subgraph can be now matched against predefined patterns or heuristics can be applied to this structure to identify the feature. In addition to topological data, geometrical data can also be represented on the arcs and nodes of the graph as attributes, which further aids the feature recognition process.

Floriani [Flor87] proposed the use of a Generalized Edge-Face Graph (GEFG) for the recognition of features such as protrusions, depressions, through holes and handles from a B-rep model. In the GEFG, faces, edges, vertices and loops are explicitly encoded, along with five mutual relationships (face-loop, loop-edge, loop-face, edge-loop and edge-vertex). This graph represents the object globally and provides multiply-connected features which can be identified and classified by partitioning the graph into subgraphs corresponding to the bi-connected and tri-connected components or features. The relationships between the various subgraphs is represented in the form of directed acyclic graph, called Object Decomposition Graph data structure. This data structure is

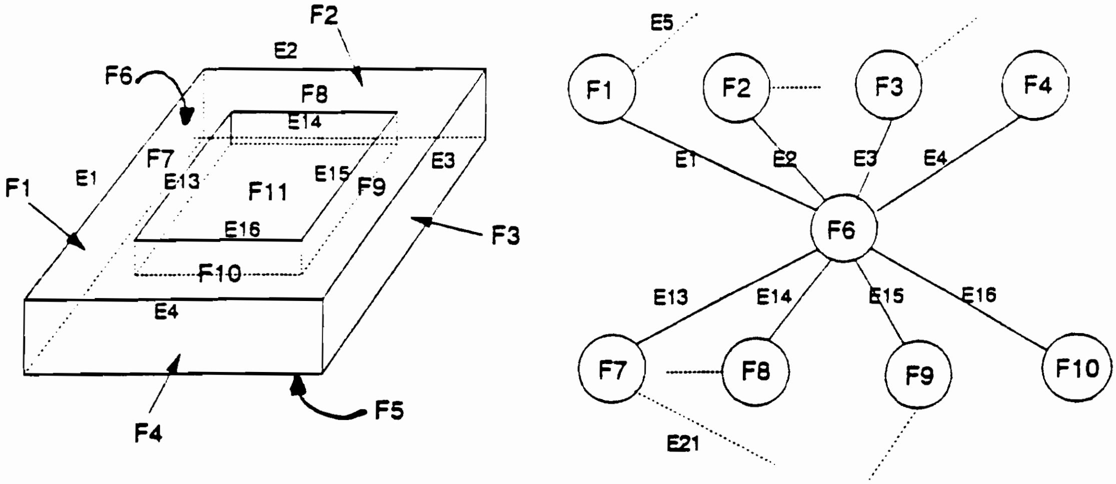


Figure 2.8 A Geometric Model and its Edge-Face Graph

a decomposition of the main edge-face graph into its features. Floriani and Bruzzone [Flor89] described an algorithm which is an improvement over the previous algorithm in terms of its ability to recognize through-holes and handles. This is achieved by an additional vertex-edge relationship.

Joshi and Chang [Josh88] developed a graph-based methodology for the extraction of features, such as slots and holes, from a B-rep model. A face-edge graph, called Attributed Adjacency Graph (AAG), is formulated based on the input from the B-rep data. Each node represents a face of the object, and each arc between two nodes represents an edge, common to the two faces represented by the nodes. Depending on the angle formed at an edge by the two adjacent faces, an attribute is assigned to the corresponding arc, convex = 1 and concave = 0. The graph is complete when all the information about the part is arranged in the graph. The methodology is limited to 2D models. In addition the methodology cannot be applied when two or more features intersect Heuristics are applied to the graph to isolate sub-graphs and to recognize the type of feature each sub-graph is representing. The implementation has been done in FORTRAN 77 on a Sun 3/50 work-station and using the solid modeler ROMULUS.

Chuang and Henderson [Chua89] developed a vertex-edge graph based methodology for the extraction of features from a B-rep model. The nodes represent the vertices and the arcs represent the edges. The topology (adjacency relationship between the edges and faces) and geometric properties (convexity and perpendicularity) of edges and faces which are adjacent to the vertex is used to classify the vertex. Thus the model is represented using a vertex-edge graph, labeled with classified vertices. Sub-graphs which represent regional shape patterns are generated and matched against pre-defined patterns for identification of features. The system has been implemented using the solid modeler ROMULUS and the PROLOG language.

Gavankar and Henderson [Gava90] proposed an algorithm for the extraction of protrusions and depressions from a B-rep model. An edge-face graph, wherein the nodes represent faces and the arcs represent edges is searched for faces with multiple edge loops. An object having a face with multiple edge loops is a one-connected graph. A protrusion or depression feature will result in a multiple edge loop, and hence a cut node, the removal of which results in the creation of subgraphs. Each subgraph represents a feature such as a protrusion or depression.

2.2.2.3 Expert System Approach

An Expert System (ES) is a computer program designed to emulate the problem solving ability of a human expert, and consists of three main components: a knowledge base, an inference engine and a user interface [Dym91]. The knowledge base consists of the facts, procedural rules and heuristics (knowledge). The two most common methods of knowledge representation schemes are the production rules and frames. Rule-based representation is the most widely used method, because it is easy to construct and its production rules are similar to the "if-then" statement in conventional programming. In frame-based representation schemes, knowledge is organized in frames or classes, which are abstracted in a hierarchy that provides for inheritance. Attributes assigned to each frame apply to both the frame as well as its descendant frames. The descendant frames can be further specialized by adding additional attributes. The inference engine defines the direction of knowledge base search, and uses one of two basic search approaches; data-driven or forward chaining, and goal-driven or backward chaining. In forward chaining, data is matched against the rules to determine whether they should be fired if satisfied. Backward chaining follows an inverse procedure by starting with a goal and moving backwards to establish relevant facts. The user interface allows for data input and output, and can provide explanations of the reasoning process.

Henderson and Anderson [Hend84] developed an expert system, called FEATURES, for the extraction of form feature such as holes and slots. The system consists of three modules: a feature recognition, a feature extraction and a feature graph construction module. Faces, edges, and vertices are extracted from the B-rep model and these are represented as facts. These facts are searched and rules of logic are applied to detect the presence of features. The next step is the extraction of the features recognized in the previous stage. The last stage is the construction of the feature graph, which is linked according to the adjacency of the features. Features which have a common entrance face are linked directly to the main node whereas the other features are linked according to their adjacency with the features already linked to the main node. The algorithm is implemented in PROLOG and uses the solid modeler ROMULUS.

Henderson and Chang [Hend88] developed an automatic process planning system which uses a feature extraction system called Feature Recognizer and Process Planner (FRAPP). The input to the FRAPP system can be either a CSG or B-rep model. The system is restricted to extracting features such as slots and holes. The feature recognition process consists of recognition, extraction, analysis, computation and compilation. In the feature recognition phase, the object data is represented as facts which are then matched against the necessary conditions for a hole or a slot definition. If any feature is identified, the face-edge list for the feature is stored. Features can be combined to form macro-features if a constraint on the sequence in which the features are to be machined must be maintained. Analysis and computation deals with deduction of the parameters of the features. During compilation the features and their attributes are arranged in a frame-based representation scheme for process planning purposes. The system is built using the KEE expert system shell and the LISP language on a TI-explorer, and the solid model ROMULUS for model creation.

2.2.2.4 Other Approaches

Falcidieno and Giannini [Falc87] proposed a hierarchical organization of Face Adjacency Hypergraphs for the extraction of prismatic features using graph theory and syntactic pattern recognition techniques. The Face Adjacency Hypergraph is a relational boundary model, in which nodes represent faces and arcs represent relationships among the faces induced by sets of edges and vertices. The algorithm operates in three steps - feature recognition, feature extraction and feature organization. Feature recognition is done using syntactic pattern recognition techniques to recognize depressions and protrusions. In feature extraction, the algorithm adds dummy surfaces to the recognized faces of the features to make the resulting object conform to Euler's formula. This is done to determine the amount of material to be removed by machining. In the organization stage, the features are arranged in a hierarchical graph with nodes representing the features and the arcs representing the relationships between them.

Perng, Chen and Li [Pern89] developed a procedure for the extraction of machining features such as slots, pockets, steps, protrusions, and holes from a CSG model. The algorithm converts the CSG model into a destructive solid geometry where all union operations are converted to difference operations so that the extracted features represent machining volumes. The extracted features are arranged in a tree structure representing machine volumes. The implementation is in the C programming language on a Masscomp 5500 computer.

Among the research efforts that have used pattern recognition, [Fu82 and Stal83], the input has been solid model data. In both the cases, the pattern representation has been a 2-D representation of the 3-D data. This leads to loss in depth information, and is not suitable for CAPP purposes. Moreover, pattern representation of geometries which are not symmetric around an axis is very complex.

In the expert systems approach [Hend84 and Hend88], the object geometries are represented as facts, and rules are utilized to determine the presence of features. The complexity of the rules is dependent on the level of geometric data. Geometric data at a low level of abstraction may lead to complex rules. CSG models provide geometric data at a higher level of abstraction, but suffer from the problem of ambiguities and cannot be used with ease in rules.

Graph theory technique has good potential in feature recognition problems when used in conjunction with B-rep and related kind of data [Li88, Flor87, Flor89, Josh88, Chua89, and Gava90]. One advantage is that the analysis can be done independently of the geometry. At the same time, as a graph models the topology in terms of the geometric primitives, geometric information can be explicitly obtained. However, if geometrical data is also captured in the graph model, the representation can be an extremely powerful aid in feature recognition and process planning. In all of the above feature recognition techniques, the complexity of the algorithms required is extremely high when there are two or more intersecting features. Also, a feature can be completely eliminated by subsequent features. This eliminates the use of feature recognition techniques for machining analysis, since all modified feature and geometry information is required for DFM analysis and process planning.

2.2.3 Feature Based Design

CAD systems represent part data in the form of entities such as points, lines and surfaces. While CAM systems require part data representation at a higher level of abstraction, such as features and the spatial relationships between them. In addition, when designers try to meet a functional requirement, they think in terms of the features or the forms that the product must have. Feature based CAD systems allow the designer to work at this higher level of abstraction than at the low level of geometric entities. In this technique, the designer uses features to model the part and thus the feature level information is explicitly represented. It provides a means for building a complete database at multiple abstraction levels, and allows the designer to associate geometric entities together and to attach non-geometric attributes to such entities. Techniques such as process planning, DFM and DFA which require reasoning based on features, can be automated when using a feature based design system. However, it is worth mentioning here that there is no comprehensive set of features that can be applied to all phases of product development. Design features are different from manufacturing features. For instance, in Figure 2.9 the design perspective could be to see the features defined as ribs, while the manufacturing perspective is to see slots. In addition, the casting process may see this feature as ribs, while the metal removal process may see slots. It is also possible that a part created using design features may not be producible using a given manufacturing process [Jone91]. This problem can be addressed either by mapping design features to the corresponding manufacturing features or by using manufacturing features for design.

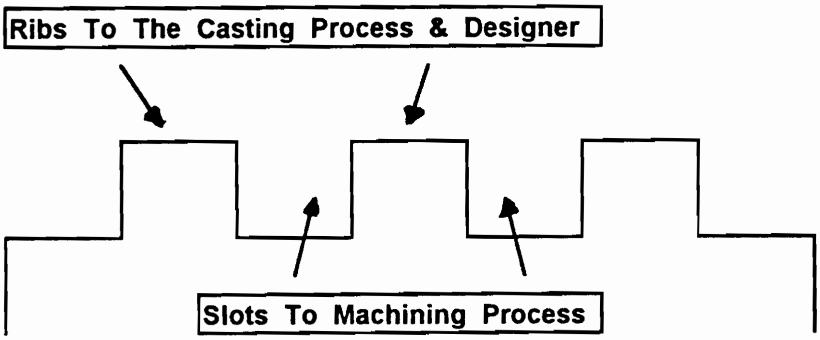


Figure 2.9 Feature Definition Varies

Dixit [Dixi88] described the MICADEX expert system which allowed machinability data to be integrated with a CAD system. The system was capable of retreiving information from a 2D CAD system and make decisions on the machining parameters. This system is limited to turning operations and the data input is 2D which further limits the scope of the software.

Laakko et al. [Laak90] described a feature-based modeling system which incorporated manufacturability analysis and process planning for prismatic objects for a 3-axis milling machine. The manufacturability analysis was restricted to individual features and is based on size and orientation only. In addition, the analysis was limited to the rules for the 3-axis milling machine. It also did not consider the spatial relationships between features.

Other researchers [Chan90, Tons94, Chen94] have used feature-based models for process planning and NC path generation. Chang [Chan90] described the QTC system which converts design features to manufacturing features to develop NC code and process plan. Tonshoff et al. [Tons94] used a language description output from a feature-based modeler to develop process plans. The process plans were geared towards a single machine with drilling and processing capabilities. Chen and Leclair [Chen94] described a method for setup generation and sequencing based on the tool approach direction and tool type. Tolerances were not taken into consideration. All of the three works described above assumed manufacturing feasiblity for the part design and do not perform any further analysis on the process plan.

2.2.4 Product Data Exchange Standard

The PDES form-feature-information-model is described in this section. PDES is a neutral file directed to data transfer between different CAD software(s). The shape representation makes use of five models: Form-feature, nominal shape, geometry, topology, and dimensions and tolerances. Features may have explict (e.g., thru_slot, blind_hole) and implicit representation (e.g., passage and protrusion). The dimensions and tolerances model allows location, size, angle coordinate and geometric information. The nominal shape model support solid models, wireframe and surface representation. The geometry model supports axis, point, vector, curves etc. The entities supported in the topology model include vertex, edge, and face. PDES being a standard for data transfer does not address the issue of representation of manufacturing information. The STEP standards are aimed at providing mechanisms for design and manufacturing data transfer. However, the representation of spatial relationships, tool accesibility, etc. are still not addressed by the STEP standards.

2.2.5 Spatial Relationships Between Features

This section conducts a survey of work done in the area of representation of spatial relationships between features.

Liu and Nnaji [Liu91] proposed the use of spatial relationships for use in assembly analysis. The spatial relationships are used to determine the degrees of freedom between components and to describe the configuration of an assembled product. Mating frames are used to represent the mating positions, which in turn can be used to determine the assembly location. The spatial relationships developed are exclusively for assembly analysis and are not generic enough to be applied to other processes.

Da Silva, Wood, and Beaman [Dasi90] investigated the explicit representation and manipulation of spatial relationships between three-dimensional features, to determine tool accessibility. They developed a language abstraction for describing interfacing and interfeature relationships for a limited set of mechanical feature primitives. The language consists of a list of lexical icons (bottom, side, top and end of a feature) and mathematical concepts (planar, coplanar, offset, parallel, orthogonal, collinear, and angular). The representation could provide ambiguous data due to the lack of a local coordinate system associated with the feature and due to the lack of an initial locating point for each feature. The relationships are restricted to regular relationships (e.g., parallel, and coplanar). The ambiguity in the definitions of relationships have to be resolved and scope of the relationships have to be expanded to completely capture the feature relationships.

2.3 Summary

Most of the research efforts on DFM has been directed towards net-shaped processes. There have been only two DFM methodologies, one for welding and the other for sheet metal. There is no formal procedure or methodology for performing the DFM analysis for machined components. Without the development of a methodology, a comprehensive effort towards the DFM analysis of machined components cannot be directed.

DFM automation efforts have been very limited in scope and focused towards one or two machines. Before automating DFM procedures, an appropriate DFM methodology has to be developed along with the identification of the data required and a data representation framework. DFM analysis of machined components require process planning information as well. Computer aided process planning research work does not perform a DFM analysis and assume a feasible design as input to the process planning module. Also, the process plan is represented as text data which does not allow for further analysis. The issue of representing manufacturing data that is required for performing the DFM analysis and the process plan have to be addressed in order to automate DFM analysis. In the next chapter the DFM methodology created and its data representation framework are described.

CHAPTER 3

RESEARCH METHODOLOGY

This chapter details the research process that is followed for developing the machined component DFM methodology, its data representation, and software prototype. Section 3.1 provides a general framework for the integration of DFM into design followed by the details of the proposed DFM methodology for machined components. The data requirements for each phase of the DFM methodology and its proposed representation scheme are identified in Section 3.2. Section 3.3 outlines the scope of the software prototype.

3.1 DFM Methodology for Machined Components

Figure 3.1 shows the traditional design process. The design process begins with the recognition of a need. The need is translated into functional or design requirements (conceptual design) which detail in precise terms what the product should be able to do including the service conditions under which it would perform. This list of requirements could also pertain to the various product life cycle phases, such as planning, design, manufacturing, marketing, maintenance, and use. Conceptual design list is therefore the highest level of design abstraction and consists of different types of information including technical, economic, ergonomic, and legal. In the preliminary design phase, the geometry and configuration of the part is developed based on the functional specifications. The part design will include the detailed specification of geometry, dimensions, material, tolerances, and surface finish. The design analysis and optimization phase performs various analyses of the part with respect to its ability to meet the functional requirements. Finally, if no product revisions are needed from the design and analysis phase, the initial part design

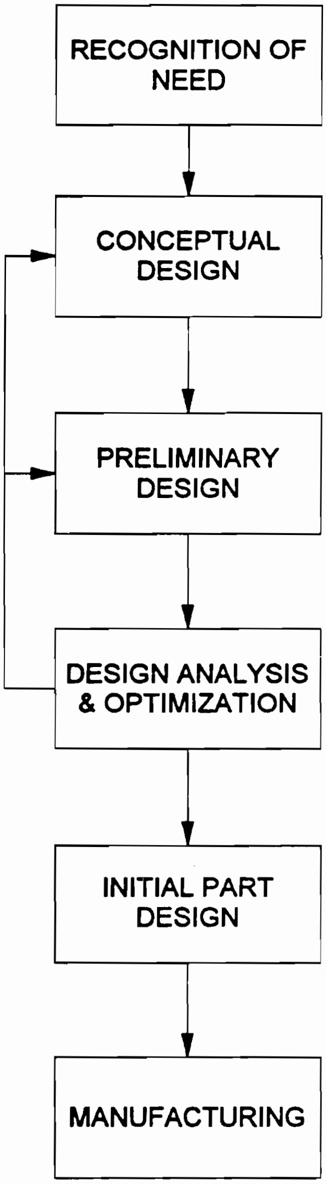


Figure 3.1 Traditional Design Process [Stol91]

is provided for manufacturing.

Figure 3.2 shows a general framework for the integration of DFM into design. The DFM feasibility and analysis module would perform the necessary manufacturing value analyses before the part is released from design to manufacturing. As indicated earlier, DFM analyses are process dependent. This is more reflected in the detailed DFM module shown in Figure 3.3. For components requiring two or more manufacturing processes for their production, the process selector/sequencer will take care of that. First the module will invoke the primary process (e.g., casting) followed by the secondary process (e.g., machining).

The DFM feasibility and analysis for machined components consists of three major phases and is shown in Figure 3.4: (i) DFM feasibility, (ii) process plan generation, and (iii) DFM analysis. The details of each module and the type of analysis to be performed at each phase are described below.

3.1.1 DFM Feasibility Module

Figure 3.5 shows the details of the DFM feasibility module which consists of six sub-modules: material analysis, dimensional checking, conventional tolerance checking, generation of spatial relationships, geometric tolerance checking, and part configuration checking. During material analysis, the material is evaluated to determine:

- if it is compatible with the process to be used (i.e., machinability),
- if it is compatible with the available tooling, and
- if a less expensive material meeting both design and manufacturing requirements can be used.

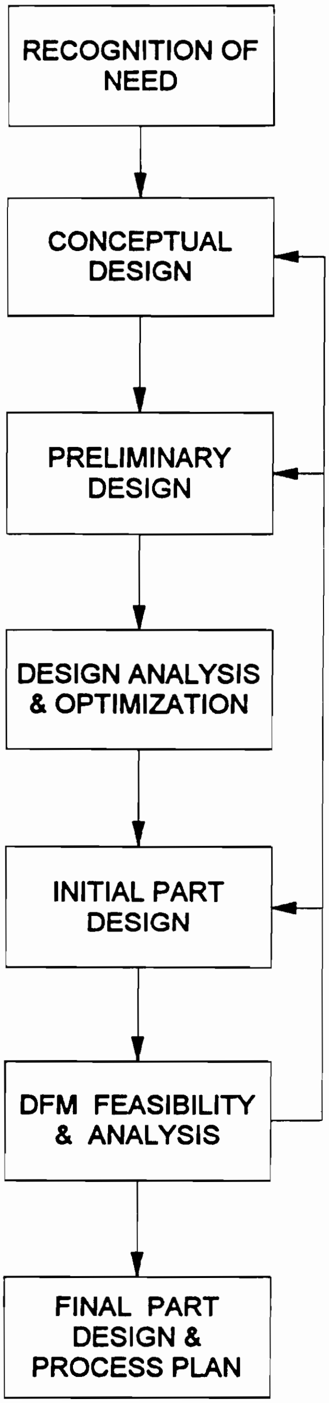


Figure 3.2 Integration of DFM into Design

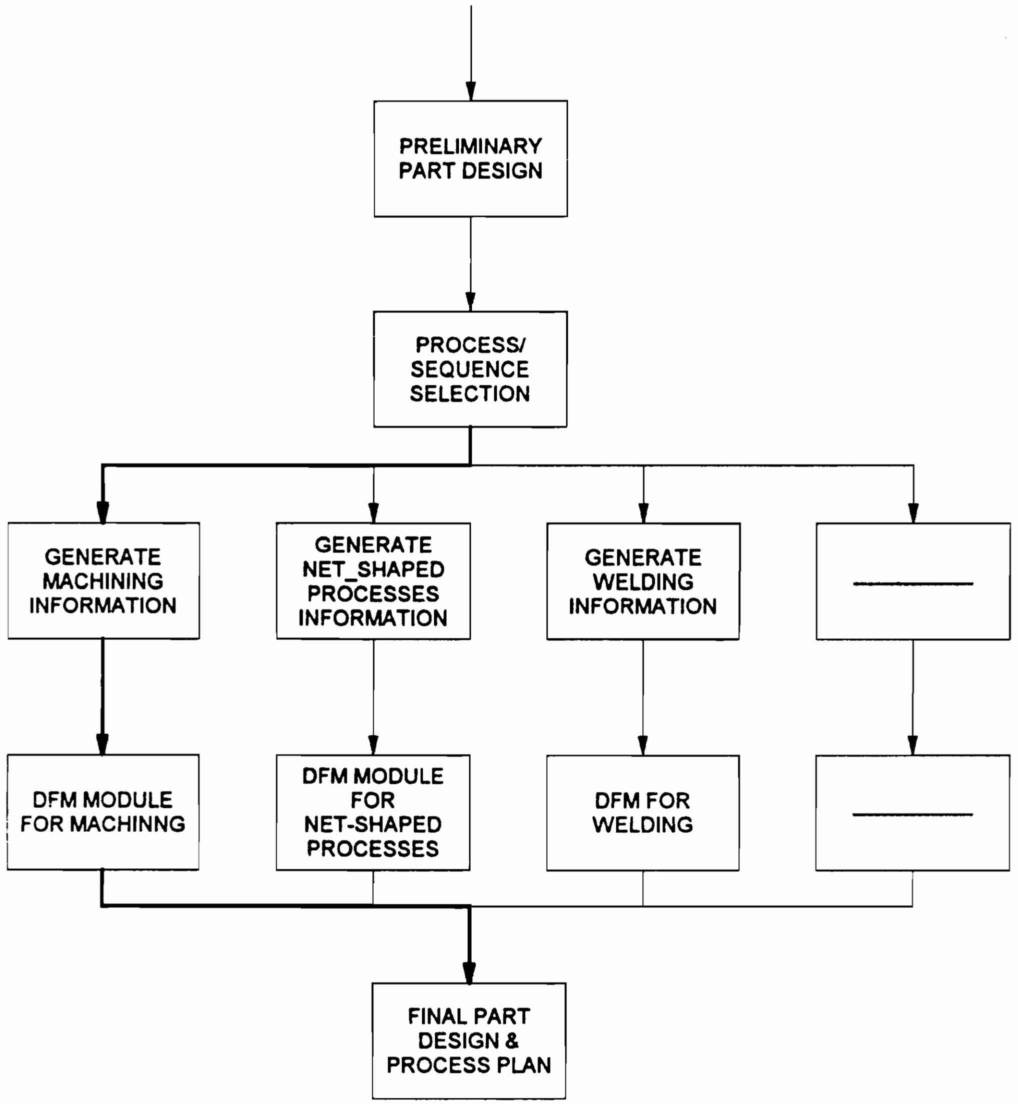


Figure 3.3 Detailed DFM Module

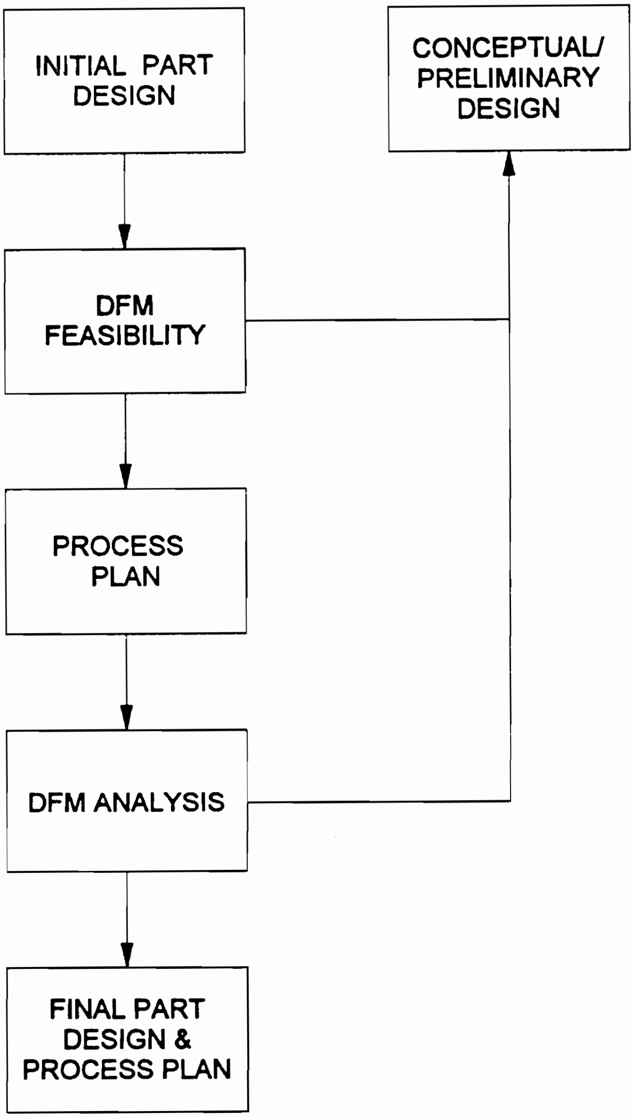
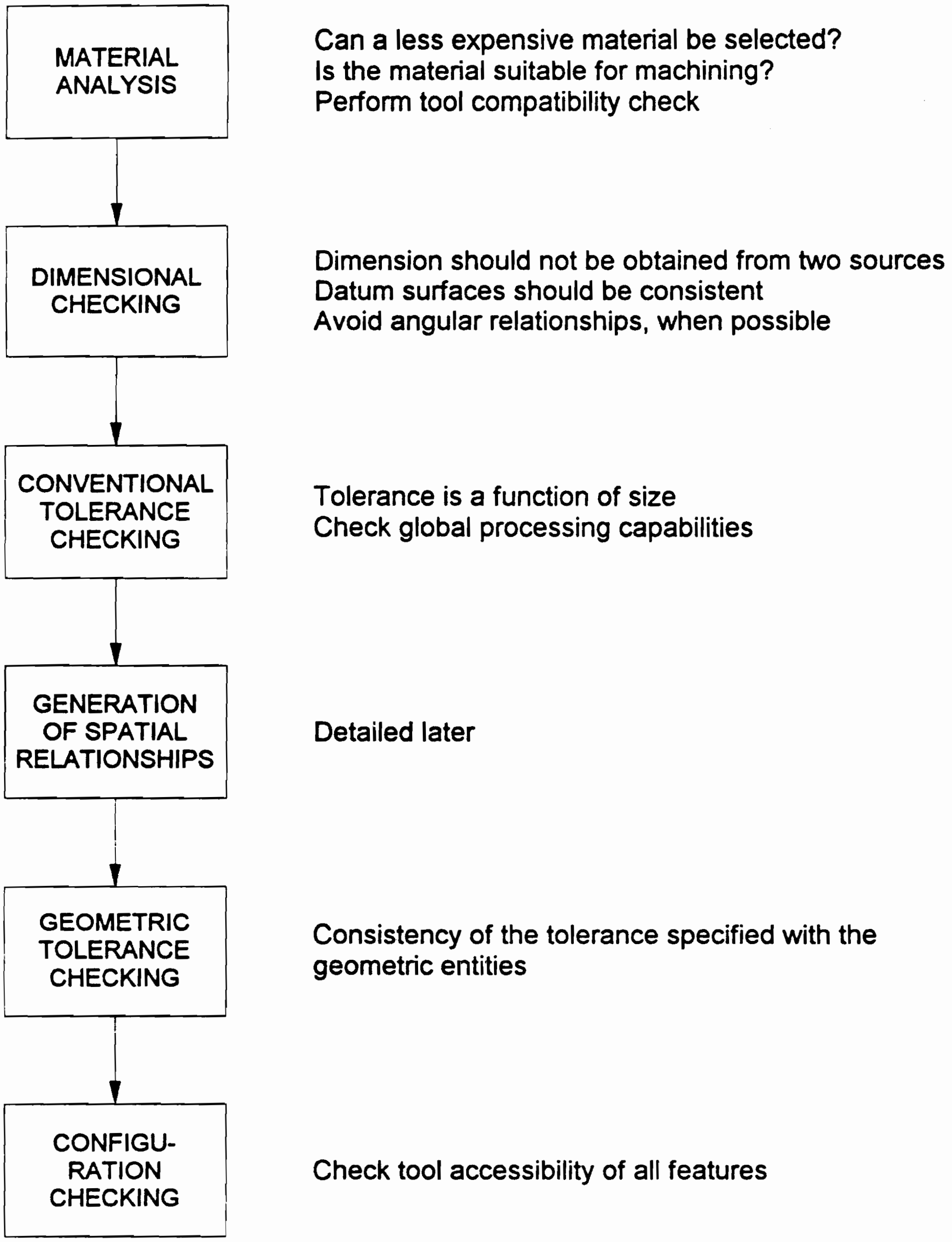


Figure 3.4 DFM Feasibility and Analysis Module



Can a less expensive material be selected?
Is the material suitable for machining?
Perform tool compatibility check

Dimension should not be obtained from two sources
Datum surfaces should be consistent
Avoid angular relationships, when possible

Tolerance is a function of size
Check global processing capabilities

Detailed later

Consistency of the tolerance specified with the geometric entities

Check tool accessibility of all features

Figure 3.5 DFM Feasibility Module

The main function of the dimensional checking module is to ensure that there are no conflicting requirements due to over-dimensioning. For example, the surface 'D' in Figure 3.6 can be obtained by either locating from surface 'A' or surface 'C'. Each method of manufacture would yield a different tolerance value and thus conflicting manufacturing requirements. Other functions to be performed by this module include checking the accuracy and completeness of dimensioning relative to the part functional requirements, ensuring minimum number of datum surfaces, and preventing the use of angular dimensioning whenever possible. The lower the number of datum's are, the fewer the manufacturing setups. The majority of manufacturing equipment provide linear movement and in turn linear dimensions are better than the angular ones.

The conventional tolerance checking module first ensures that the magnitude of the tolerance is a function of size. Other functions would include checking the magnitude of tolerances against the boundary limits of the processing capabilities.

In order to check the geometric tolerances and part configuration, the spatial relationships between features, such as parallelism, perpendicularity, etc., must be determined. The definitions of the various types of spatial relationships and their representation are provided in Section 3.3. Additionally, in the spatial relationships module, the tool accessibility directions for each feature are updated based on these spatial relationships. When a feature is defined, its tool accessibility directions are explicitly represented. (Section 3.3 provides the details of data representations). However, the type of feature and hence its tool accessibility direction can be modified due to the adjacency of another feature. For example, in Figure 3.7, the hole_A which was of the blind_hole type, has been changed to a thru_hole type due to the presence of slot_A which in turn increases the tool accessibility directions by one.

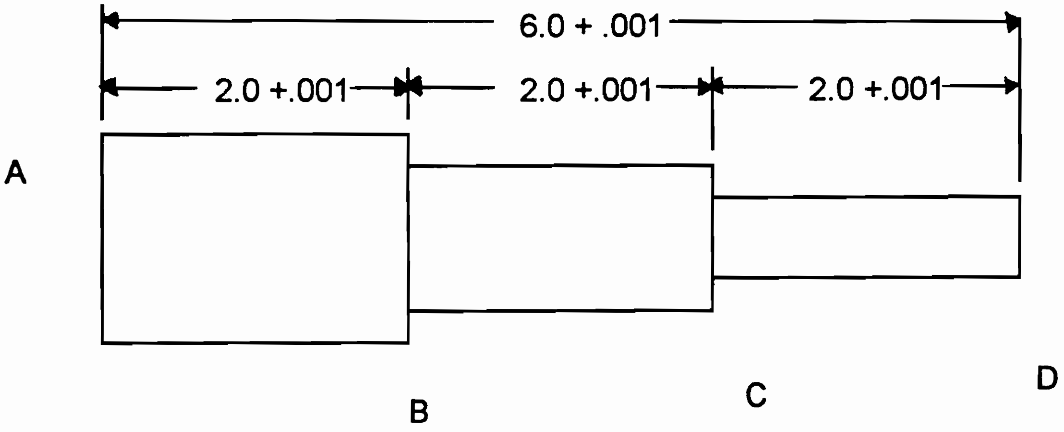


Figure 3.6 Dimension From Two Sources

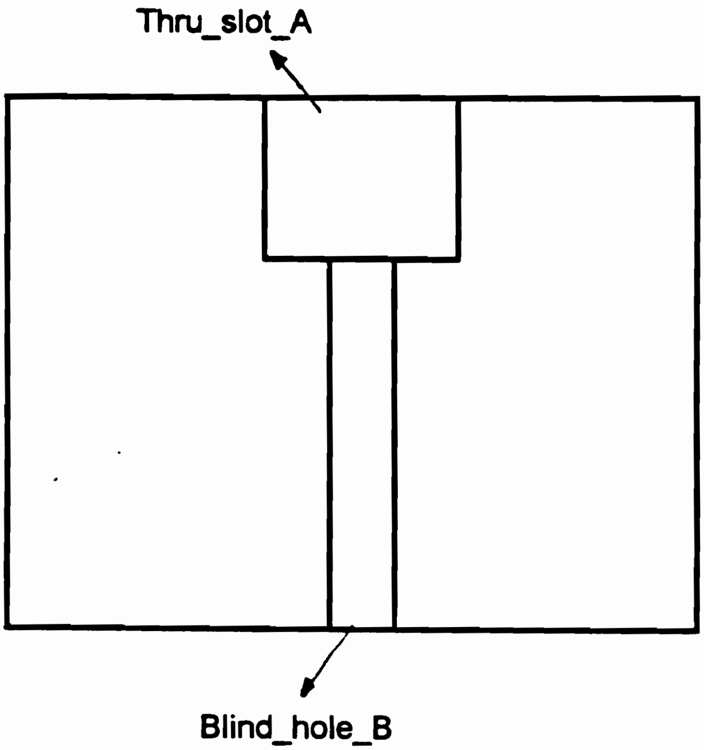


Figure 3.7 Tool Accessibility Change

In the geometric tolerance checking module, all geometric tolerances, such as parallelism, concentricity, etc., are checked in terms of feasibility and accuracy. For example, if there is a perpendicularity requirement between two surfaces and their spatial relationship is shown to be parallel, a flag will be set for the designer. Also, the magnitude of the geometric tolerance values will be checked against the maximum permissible ones, if any.

The main function of the configuration checking module is to determine the tool accessibility for all the negative features (e.g., slots, holes, and steps) in the part design using the features' tool accessibility directions and their spatial relationships. If the tool accessibility is not feasible, the designer will be notified.

3.1.2 Process Plan Generation Module

The process plan generation module assumes that the part will be machined from raw stocks. If machining is a secondary process to casting or forging in the manufacture of the part, a user interactive procedure to determine the surfaces to be generated has to be available. Each part consists of a primary or positive feature and secondary or negative features. Secondary features are defined on the primary feature. The details of feature representation and manipulation are provided in Section 3.3.

The first step in the process planning module is to determine the surfaces to be generated. This is achieved by comparing the design requirements of the primary or positive feature with that of available raw materials. If the design requirements can be met with available raw materials, the outer surfaces of the primary feature need not be machined. Else, all surfaces will have to be generated. Each surface/feature to be generated is then mapped to the appropriate process automatically. The sequence by which the surfaces will be generated is then determined using a sequence identifier algorithm which is based on the precedence requirements between surfaces. The precedence requirements include all design specifications such as dimensions and tolerances. A binary relational square matrix of all surfaces/features is first created. If a precedence relationship exists between two surfaces/features a value of one is assigned, else a value of zero is used. The steps of the sequence identifier algorithm are as follows:

1. If surface/feature S_{ij} has a precedence constraint relative to surface/feature S_{mn} (where i & m represents the rows, and j & n represents the columns of the binary relational matrix), place a 1 on the ith row and nth column.
2. Resolve reciprocal constraints as follows: if S_{ij} has a precedence constraint based on S_{mn}, S_{mn} should not have a precedence constraint based on S_{ij}
3. Compute the total precedence constraint(s) for each row.
4. All surfaces/features with a total constraint of zero can be generated. From this list, a feature is selected to determine the setup or primary access direction. It is assumed that each feature is accessed from its primary access direction unless it became infeasible.
5. Check other surfaces/features with zero constraint to determine if they can be produced with the same process and have same tool access direction. These surfaces/features can be generated in the same setup, if desired.
6. Repeat steps 1 to 5 until all surfaces/features are processed.

Once a feasible manufacturing sequence is determined, a cost efficient process plan can be developed by including other factors such as tolerance specifications, quantity required, available machinery, due date, etc.

3.1.3 DFM Analysis Module

The last phase of the DFM feasibility and analysis module is DFM analysis (see Figure 3.8) which covers tolerance control, tolerance analysis, and configuration analysis.

In machining a component, sometimes design datum(s) can not be used for locating the part, and the process planner would have to re-dimension the part design relative to the new datum(s). Additionally, some of the design requirements may necessitate the part to visit several setups to be accomplished.. In these situations, the process planner would first decide on the amount of stock (material) removal and then calculates the working dimensions. Whether nominal sizes are added or subtracted to determine new dimensions or working dimensions, their variability (tolerances) are always added. This phenomena is known as tolerance stacking. Also, from a manufacturing cost point of view, the tolerances of the working dimensions should be relaxed enough without violating design specifications. The process of analyzing the tolerances to ensure that tolerance stacking does not result in a violation of design requirements and that the tolerances of the working dimensions are as relaxed as possible is known as tolerance control. These activities will be performed by the tolerance control module.

The second sub-module in the DFM analysis module deals with a different type of tolerance analysis. Sometimes, when a tolerance value is relaxed, the need for another setup or additional processing is eliminated. This type of tolerance analysis must of course be performed relative to the design functional and performance requirements. This type of analysis can not be easily automated since it requires the automation of product functional and performance requirements.

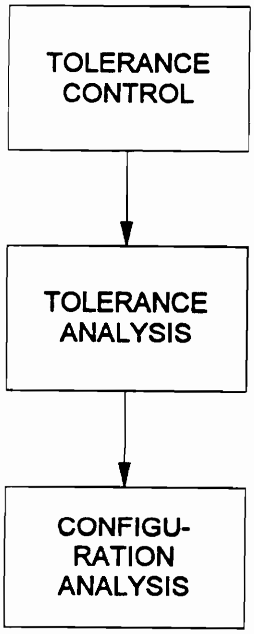


Figure 3.8 DFM Analysis Module

Finally, in the configuration analysis module, the part configuration is analyzed relative to ease of manufacture. For example, if four holes are to be drilled on the same surface, manufacturing would be easier if all of them require the same tool (i.e., same size) since tool changing time will be minimized. Other DFM analyses would include:

- reducing material removal requirements,
- using standard tools,
- reducing number of requirements,
- minimizing effect of tool action, and
- recognizing the process needs.

Figure 3.9 provides examples showing the design before and after for the above DFM configuration analyses. In addition to part configuration analysis, the process plan may also be analyzed for possible improvements. For example, in Figure 3.10 the process Plan_A will cause a drill drift since the Wedge_A is generated prior to the hole_A. By changing the sequence of processing, this can be avoided a shown in Plan_B.

3.2 Data Representation Scheme

Before the data can be represented, the data and its type must be determined. Table 3.1 shows the type of data requirements for each of the DFM modules described above. For example, for the whole DFM feasibility module, information on the part material specification, material database, part dimensions and tolerances, and general processing capabilities would be required.

The primary objective of any data representation is to allow algorithms to access and manipulate the data for the desired applications. A complete,

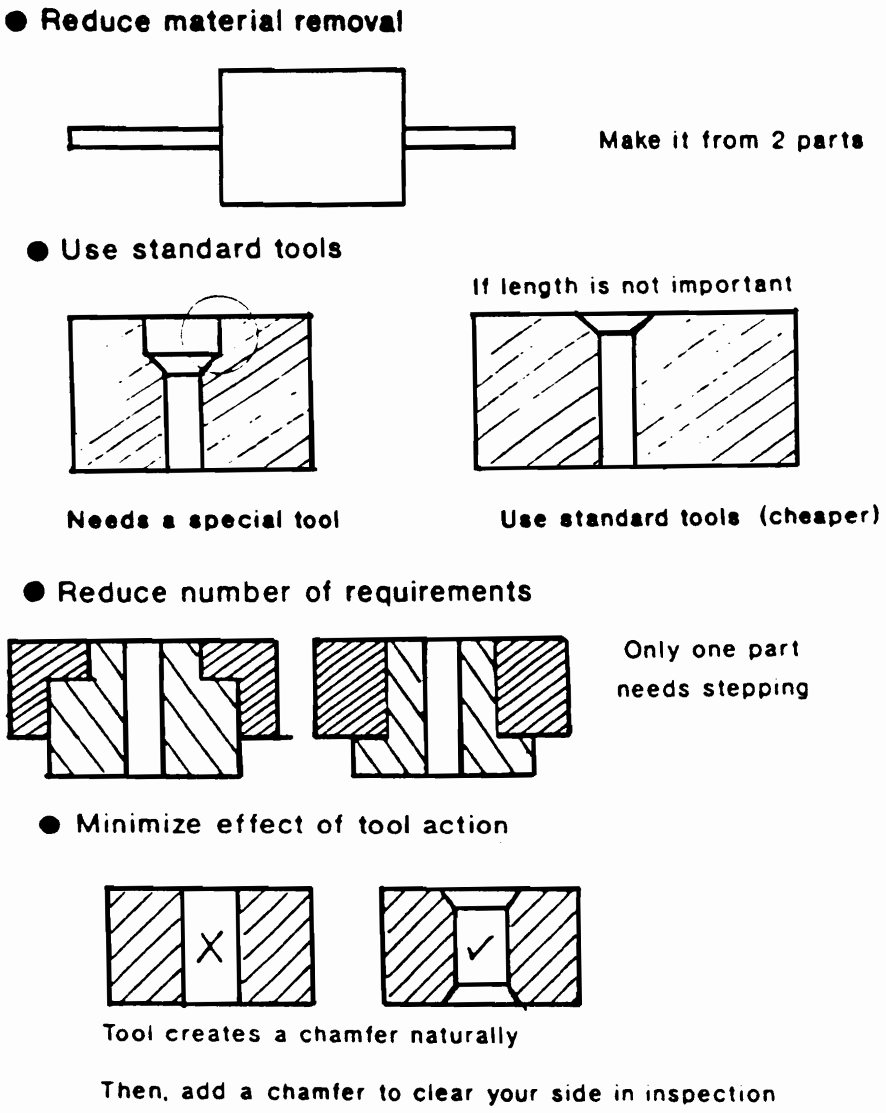
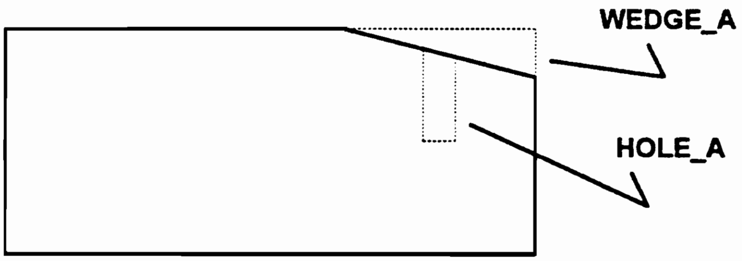


Figure 3.9 Improvement in Design



PROCESS PLAN_A	PROCESS PLAN_B
	Drill Hole_A
Machine Wedge_A	
	Machine Wedge_A
Drill Hole_A	

Figure 3.10 Improvement in Process Plan

Table 3.1 Data Requirements

DFM MODULE	DATA REQUIREMENTS
DFM Feasibility Module	
Material Analysis	Material specified, material database, tooling information
Dimensional Checking	Dimensions represented as entities and not as text, datum surfaces, part function
Conventional Tolerance Checking	Tolerances represented as entities, not as text
Generation of Spatial Relationships	Features, definitions of spatial relationships
Geometric Tolerance Checking	Tolerances represented as entities, spatial relationships between features, orientation of surfaces
Configuration Checking	Tool accessibility of features
Process Plan	
Surface Sequence Identifier	Tool access directions of surfaces and features, orientation of features, datum surfaces, dimensions and tolerances, mapping of surfaces and features to processes, available machinery database
DFM Analysis Module	
Tolerance Control	Process plan, dimensions and tolerances
Tolerance Analysis	Mapping of product function and design requirements to part geometry, process plan, dimensions and tolerances
Configuration Analysis	Mapping of product function and design requirements to part geometry, process plan, rules for analysis

standard data representation scheme for design and manufacturing application requirements does not exist since the applications are diverse and the data requirements are different. Although the most comprehensive data representation format is PDES, manufacturing information is still under represented. Nonetheless, PDES format and guidelines is used as the foundation for the geometric data representation in this research

Figure 3.11 shows the developed object-oriented framework for representing the DFM data requirements. In order to allow for software expansion to include other processes such as casting and welding, and code reusability, the shape information is separated from the process specific information. Other required information stored in databases, such as material database, are accessed through the application module as and when necessary. Separate object-oriented classes are maintained for storing each of the entities in the framework, i.e. feature, geometry, topology, dimension, tolerance, spatial relationships and process plan. These are described in detail below.

3.2.1 Data Structure

The developed data structure for the storage of information is shown in Figure 3.12. The part points to the first feature in the feature list. The feature points to its bounding faces, real and virtual. The faces point to the loops formed due to its bounding edges. The loop points to the bounding edges and the edges have pointers to the appropriate vertices. A face list is maintained for each feature. A separate face list, which consists of only real faces, is maintained for the part.

Dimensions can be specified between faces of two different features, or between faces of the same feature. It may also be specified on a single entity of

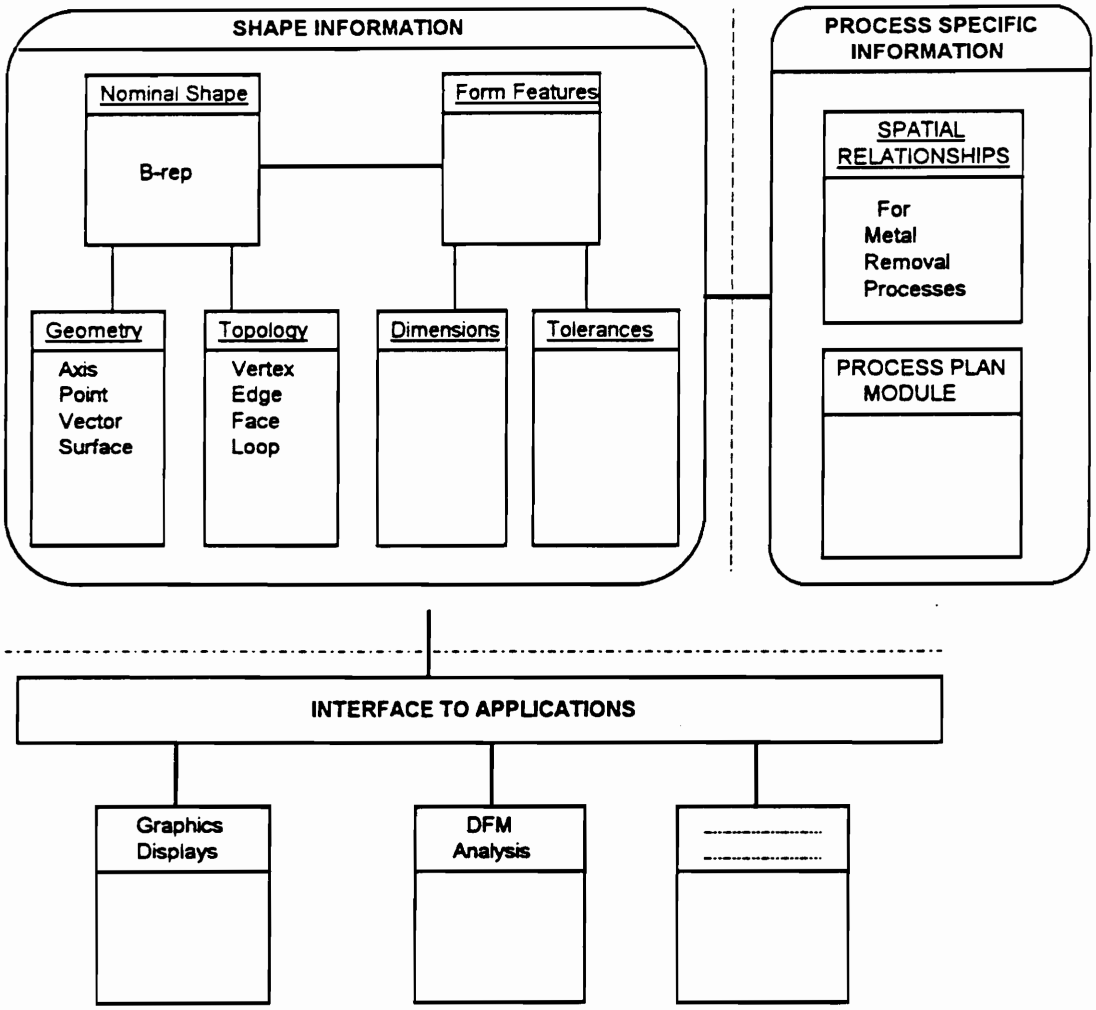


Figure 3.11 Data Representation Framework

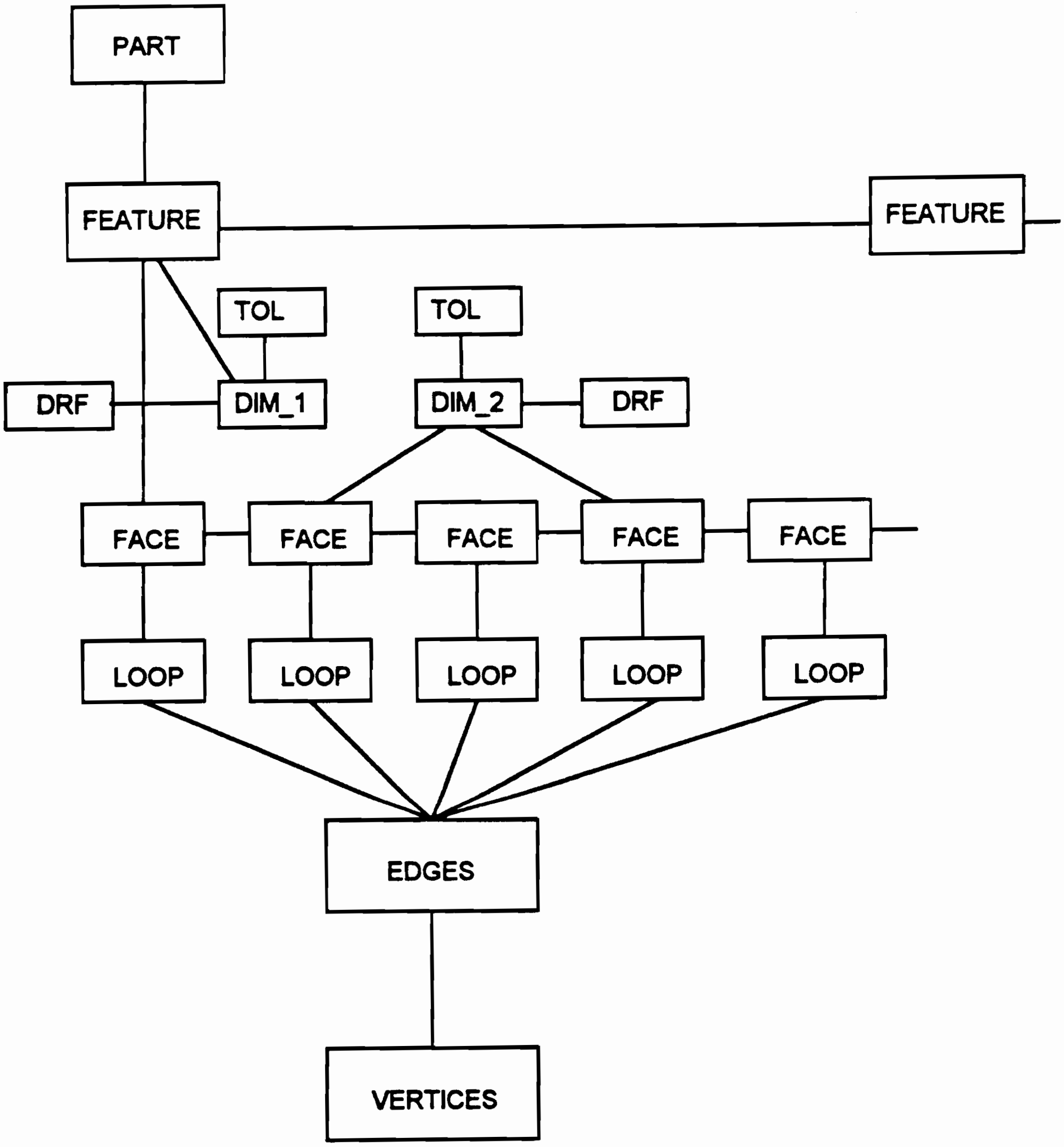


Figure 3.12 Data Structure

a feature, such as the radius of a blind hole. Tolerances may be applied on dimensions (size), as geometric tolerance between two entities (form), or on the orientation axes of a feature. Depending on the type of dimensions and tolerances, they are linked to the appropriate entities through pointers.

3.2.2 Feature Class

Features are classified as primary, secondary or tertiary. Primary features are limited to positive features, such as a brick and a shaft. Secondary features are defined on the primary feature. In this research work, both positive and negative secondary features are used. Tertiary features are design specifications/features which do not change the geometry drastically, such as, chamfer, heat treatment, surface treatment etc. They are classified into two classes: types which change geometry, and types which do not. Annealing is an example of a tertiary feature with no associated change in geometry, while chamfer and powder coating are examples which have a change in geometry.

The following are the features used in the software prototype:

1) **Primary features** Shaft; Brick
2) **Secondary features** Shaft; Slot; Step; Corner; Blind_slot; Thru_slot; Hole; Thru_hole; Blind_hole; and Groove.
3) **Tertiary features** Chamfer, Heat treatment

A sample description of a Thru_slot is provided in Figure 3.13. The following section provides the definitions of the terminology used in feature definition:

Locate_point: The initial point of a feature F_i is the origin of the local coordinate system about which the feature is defined.

Local_orientation: The local orientation defines the orientation of the local

Feature_Id	int no.
Feature_Name	thru_slot
Feature_Type	secondary
Parameters[3]	h, w, l
Local_Orient	
X-Axis	1, 1, 0
Y-Axis	0, 1, 0
Locate_Point	3,3,0
Face_list	ptr
Tool Access	0,0,1,0,1,1
Label	A

Figure 3.13 Description of a Thru_Slot

coordinate system used by the feature relative to world coordinates.

Tool Access Directions: The tool access directions for a Feature F_i are defined as vector directions along which the feature may be accessed. The tool access directions are included in the feature definition. The maximum number of tool access directions will depend on the type of feature. The tool access directions are hierarchically specified, so that the primary accessibility direction will be considered for process planning. All tool access directions are specified in relation to the coordinate system of the feature. Some features will have faces which provide partial tool accessibility. Partial tool accessibility occurs when a virtual face which does not provide complete access to the feature exists. In this case, the virtual face is smaller than the opposite face or the virtual face and its opposite face are skewed. For instance, consider the keyway and hole in Figure 3.14. The hole can be accessed because of the partial accessibility definition. However the keyway cannot be completely accessed through the virtual face.

Label: The label is a character used to name a feature for identification.

3.2.3 Geometry Class

The geometry class, consists of subclasses - point, vector, surface, and axis. These classes are used to store the geometric information. For instance, the point class has the x, y, and z coordinates of the point as members of the class. In this research work, vectors are restricted to be line segments and surfaces are restricted to be flat planes. Point, surface, vector and axis can be specified as real or virtual. Virtual entities are those used for feature definitions, but are not part of the part design. Virtual entities are useful in checking tool accessibility.

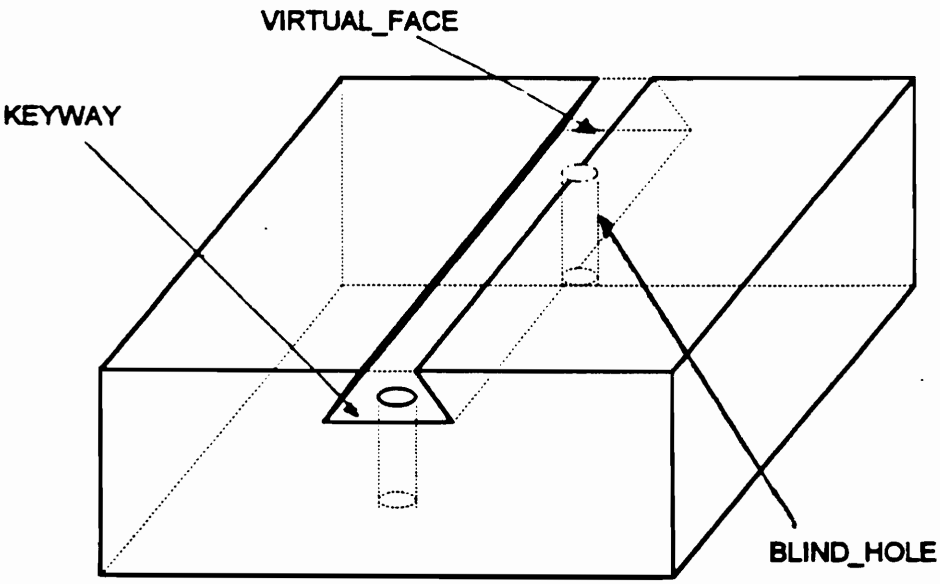


Figure 3.14 Example for Partial Accessibility

3.2.4 Topology Class

The subclasses in the topology class are: face, loop, edge and vertex. All these classes are used to maintain a linked list of the appropriate entities. The vertex class, for instance has a pointer to a point entity and point to the previous point, as well as the next point. Similar linked lists are maintained for all the entities.

3.2.5 Dimension Class

The dimensions class is shown in Figure 3.15. Dimensions class is derived further as linear, radial, angular, and other class. The dimensions class allows the explicit representation of user-defined dimensions and can be accessed through the topology class or through the features class. Dimensions can be considered to be the length of an edge (distances between two vertices), or as the distance between two faces. The two entities can belong to the same feature or to different features. The entity which is used as the datum is explicitly identified (using pointers). The structure for dimension stores not only the value, but also pointers to the appropriate entities (faces, edge and vertices). Tolerances are attached to the dimensions and features.

3.2.6 Tolerance Class

The tolerances class is shown in Figure 3.16. Conventional and geometric classes are derived from the base class. Conventional class is further split into linear, radial, and angular. The geometric class is further derived as perpendicularity, parallelism, concentricity etc. Depending on the type of tolerances, they are attached to features/surfaces or to dimensions. Tolerances on size, are attached to surfaces. Orientation tolerances (e.g., perpendicularity, and parallelism) are attached to features as well as faces.

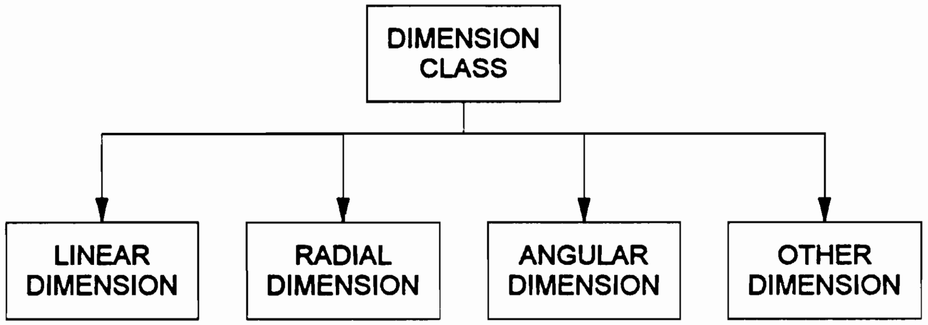


Figure 3.15 Dimension Class

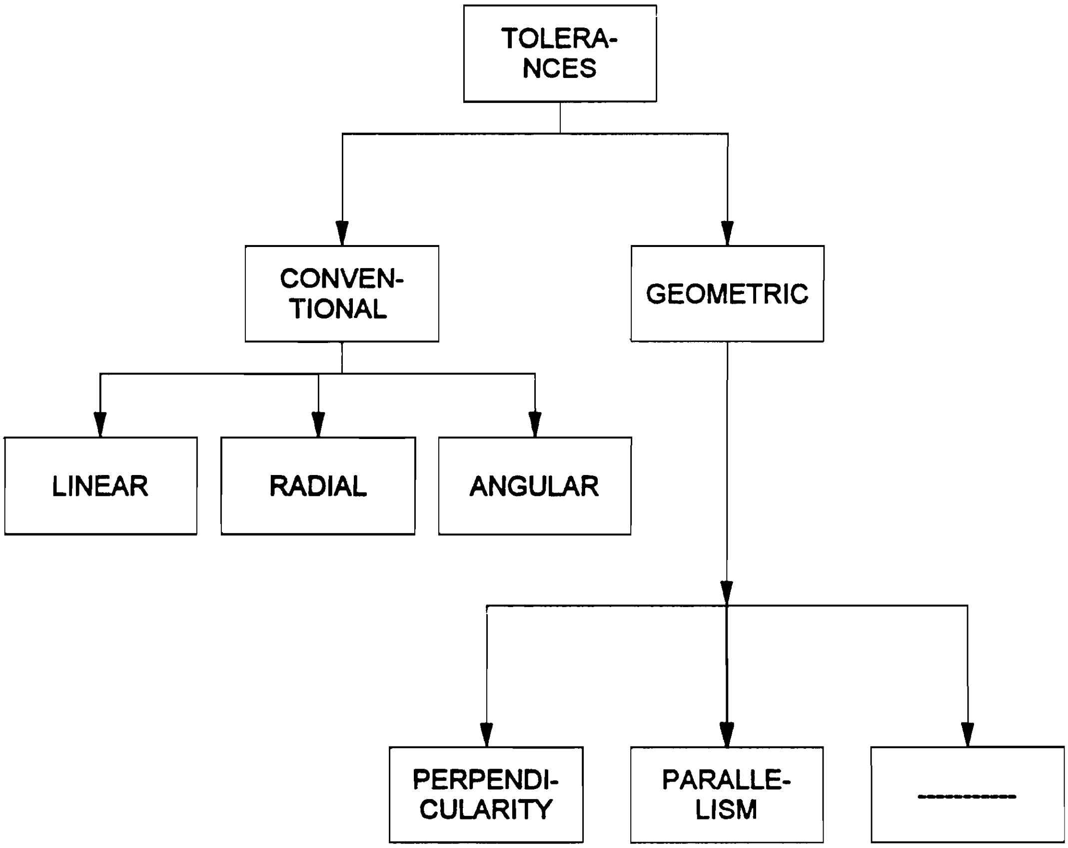


Figure 3.16 Tolerance Class

3.2.7 Spatial Relationships Class

The spatial relationships class is a data class for storing the spatial relationships information. Figure 3.17 shows the sample definition of a spatial relationship. All the relationships between features are directed. One of the features is used as the datum/host_feature relative to which the other feature (guest feature) is measured. The definitions of the various spatial relationships are as follows:

Perpendicularity Definitions: Perpendicularity is defined with reference to the three axes of the feature. These are:

Perpendicular_x: Feature F_i and feature F_j are said to be perpendicular_x, if the angle between the x-axis of F_i and the x-axis of F_j is 0, but the angle between the y-axes is 90^o (also implies that the angle between the z-axes are at 90^o). Perpendicularity is specified by a vector [+1, 0, 0] to indicate perpendicularity about the x-axis. The positive sign preceding the 1 indicates that the coordinate axes of F_j is clockwise to that of F_i.

Perpendicular_y: Feature F_i and feature F_j are said to be perpendicular_y, if the angle between the y-axis of F_i and the y-axis of F_j is 0, but the angle between the x-axes is 90^o (also implies that the angle between the z-axes are at 90^o). Perpendicularity is specified by a vector [0, +1, 0] to indicate perpendicularity about the y-axis.

Perpendicular_z: Feature F_i and feature F_j are said to be perpendicular_z, if the angle between the z-axis of F_i and the z-axis of F_j is 0, but the angle between the x-axes is 90^o (also implies that the angle between the y-axes are at 90^o). Perpendicularity is specified by a vector [0, 0, +1] to indicate perpendicularity about the z-axis.

Parallel: Feature F_i and feature F_j are said to be parallel if the angles

relation_id	int
From_featureid	int
To_fea_Id	int
From_fea_name	thru_hole
To_fea_name	blind_hole
From_fea_type	secondary
To_fea_type	secondary
Perpendicularity	int [1,0,1,1]
Parallel	int 0
Co_planar	int 0
Co_linear	int [1,0,1,1]
Adjacent	int [0,0]
Offset	float [ ]
Angular	float [ ]

Figure 3.17 Specification of Spatial Relationships

between the x, y, and z axes of the two feature coordinate systems are 0 or if the vectors representing the x, y and z axes of the two local coordinate systems are the same. This relationship is specified by a 1 (parallel) or 0.

***Co-planar*:** Feature F_i and feature F_j are said to be co-planar, if the origins of the local coordinate systems are on the same plane, and the vector representing the Z-axes directions are the same. This relationship is specified by a 1 (co-planar) or 0.

Co-linearity: Co-linearity is defined about the x, y and z axis of the features. These are:

Co-linearity_x: Feature F_i and feature F_j are said to be co-linear_x, if $y_i = y_j$, and $z_i = z_j$, and $x_i \neq x_j$ and the vector representing the x-axes are the same.

Co-linearity_y: Feature F_i and feature F_j are said to be co-linear_y, if $x_i = x_j$, and $z_i = z_j$, and $y_i \neq y_j$ and the vector representing the y-axes are the same.

Co-linearity_z: Feature F_i and feature F_j are said to be co-linear about the z-axis, if $x_i = x_j$, and $y_i = y_j$, and $z_i \neq z_j$ and the vector representing the x-axes are the same.

The co-linearity is specified by a three field data. For instance, "1,0,0" indicates co-linearity_x.

***Co-accessible*:** Feature F_i and feature F_j are said to be co-accessible, if they are accessible from the same direction. This relationship is specified by a 1 (co-accessible) or 0.

Adjacent: Feature F_i and feature F_j are adjacent if a face f_{ik} of F_i and a face f_{jl} of F_j is coplanar, and the faces intersect. This will lead to four possible interactions:

- face f_{ik} forms a loop on face f_{jl}.
- face f_{ik} is the same as face f_{jl}.
- face f_{jl} forms a loop on face f_{ik}.
- face f_{ik} and face f_{jl} intersect but neither forms a loop on the other.

This information is important in determining tool accessibility changes. This relationship is specified by a four field parameter. A 1 (adjacent) or 0 in the first data field represents whether it is adjacent or not. The second and third field stores the id's of the faces of the host feature and the guest feature respectively which are in contact. The fourth data field identifies the type of interaction between the two faces.

Offset Distance: Offset distance is a three field data which describes the distances between the locate_points of the two features along the x, y and z axes of the primary feature.

Angular: Angularity is a three field structure, which is used to represent the angles between the co-ordinate systems of the two features. This relationship is determined by transforming the coordinate axes of the guest feature to the coordinate axes of the host feature. The x and y axes of the guest feature are specified in terms of the coordinate system of the host feature.

3.2.8 Process Plan Class

Process plan should not be represented as text entity, since the DFM analysis module uses the process plan. The process plan module stores the sequence in which features are machined. Figure 3.18 shows an instance of a process planning class. The setup number, orientation of the raw material, surfaces generated, features generated if any, the process type and the machine

Step_id
Setup_num
Option_num
Orient_xaxis
Orient_yaxis
Surfaces_gen
Feature_gen

Process_type
Machine_type
Machine_num

Ptr1 (pointer to previous step)
Ptr1 (pointer to next step)

Figure 3.18 Specification of Process Plan used are defined in this class. An integer variable called option number is used to store the number of other surfaces/features that could be generated instead of that feature. This is useful in generating alternate sequences. Pointers to the previous step and the next step are also provided.

In the following chapter the DFM methodology is illustrated with the use of two case studies and in Chapter 5 the software prototype operation and three examples are described.

CHAPTER 4

CASE STUDIES ON DFM METHODOLOGY

This chapter illustrates the manual DFM methodology developed by presenting two case studies. In both the cases, manufacturing considerations are analyzed manually while keeping in focus the design function.

4.1 CASE STUDY 1 - Drill Sleeve

In this case study, the manual DFM analysis of a drill sleeve is performed. The initial part design submitted for manufacturing analysis is shown in Figures 4.1 and 4.2. The function of a drill sleeve is to support a drill. Based on this function, the design requirements for such a part are:

- Ability to absorb shock loads
- Wear resistance, (hence hardness should be high for the material)
- Simple to operate

The drill sleeve needs to be tough enough to withstand the cutting force during a drilling operation. The taper on the drill sleeve is used to hold a drill bit and to be held by the drill machine. The surface of the drill should be hard enough to preserve the amount of taper, both internal and external, and the surface finish of the taper. And, the drill sleeve should be tough enough to absorb the vibration loads that occur during drilling.

4.1.1 DFM Feasibility Analysis For Drill Sleeve

In this section a DFM feasibility analysis is performed on the drill sleeve design to check for initial manufacturing feasibility. The feasibility check is

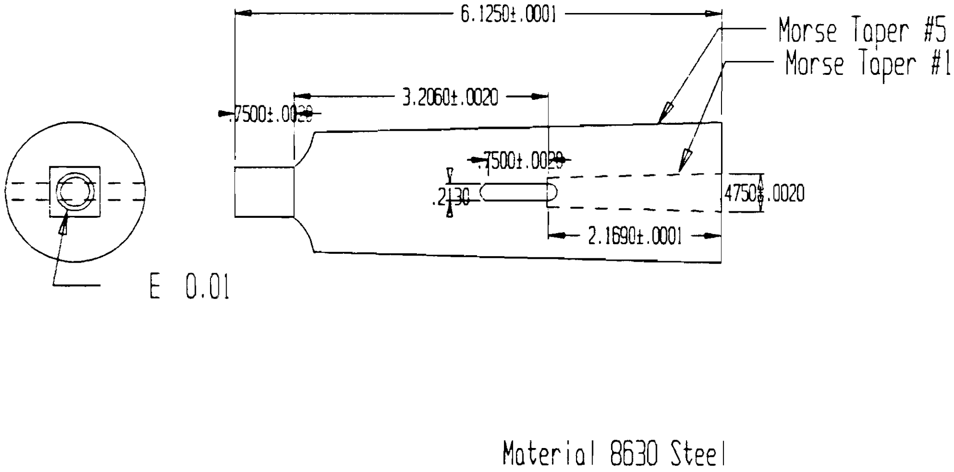


Figure 4.1 Initial Drill Sleeve Design with Dimensions

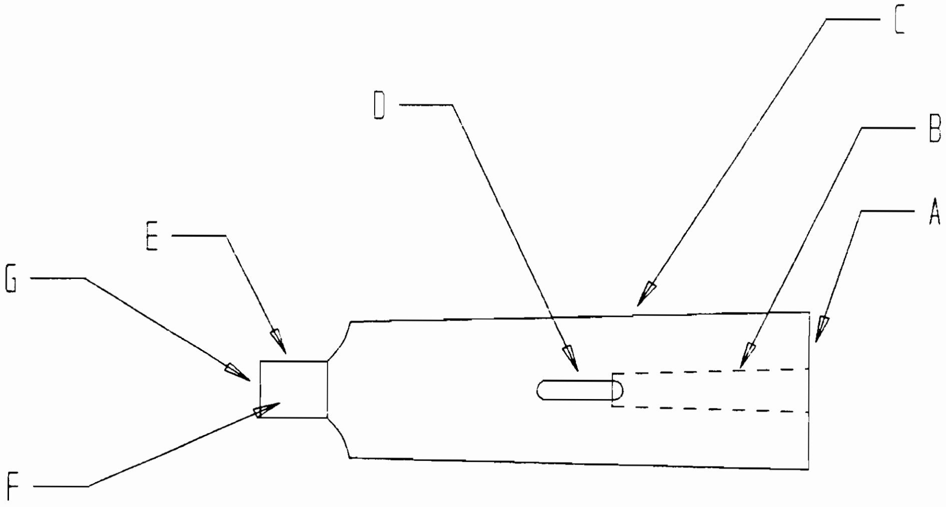


Figure 4.2 Initial Drill Sleeve Design with Labels

performed with respect to materials used, dimensions specified, conventional and geometric tolerances assigned, and part configuration.

Material Analysis: Material selected by the designer is alloy steel - 8630. This material meets all the functional requirements of a drill sleeve. The material is tough and resistant to wear (Hardness 187 Bhn). However, the cost of the material is high and the machinability rating (40) is very poor. Since the part is to be machined, the material is definitely not appropriate. An alternate material, 1018 steel is selected. 1018 steel has better toughness as compared to 8630. The lower amounts of carbon provide improved machinability (machinability rating - 70). The hardness value is not very high (120 Bhn). Carburization of the drill sleeve can provide the necessary hardness for the surface.

Dimensional Checking: In dimensional checking, various dimensions are analyzed to determine datum's, and proper constraining of dimensions. The preliminary part design has been overdimensioned. For example the dimension marked 3.2060 +/- 0.002 can be obtained from two different sources. In addition there are too many datum's as a result of over dimensioning. Using too many datum's results in manufacturing errors and unacceptable parts. The functional requirements of the part does not require the specification of two datum's. All unnecessary dimensions are eliminated.

Conventional Tolerance Checking: The tolerances specified on the preliminary design are too tight. For instance, the tolerances on the dimension 6.125 +/- 0.0001 is too tight for turning on a lathe. The tolerance requirements for the design of a drill sleeve based on the design handbook are all standard tolerances. This specification of tolerance can be relaxed to standard tolerances.

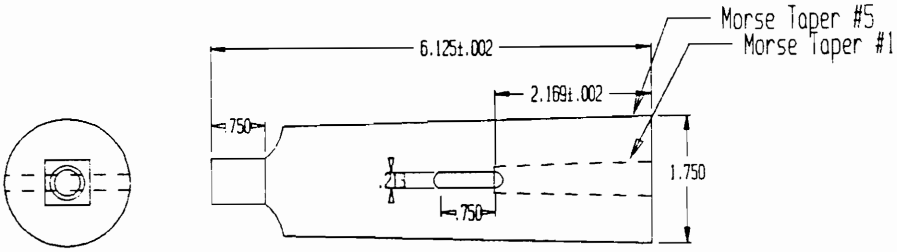
Geometric Tolerance Checking: The specifications of geometric tolerances are erroneous. Enough attention has not been paid to the labeling of surfaces. For instance the concentricity value applied to the 0.475" circle is with respect to E. However, E is a flat surface, with respect to which a concentricity cannot be applied. In addition, the geometric tolerances are not a main consideration in the design of a drill sleeve. All geometric tolerances were checked for correct labeling.

Configuration Checking: All features and surfaces are accessible and the configuration analysis is found to be okay. The part design after DFM feasibility analysis is shown in Figure 4.3.

4.1.2 Process Plan Sequence For Drill Sleeve

In this section the generation of the process sequence is elaborated for Case Study 1. Using the process sequencer algorithm the dependency matrix table is first generated (Table 4.1). Since, face A is the datum with respect to which all other surfaces/features are dimensioned, A has to be machined first. Surfaces/features B, C, D, and G are dimensioned with A as the datum and hence require A to be machined before they can be machined. E and F cannot be machined unless G is machined, since they are dimensioned with respect to G. Heat treatment is for all the surfaces and can be done only after all the surfaces have been machined.

A process plan sequence based on this dependency table is given in Table 4.2. Surface A is machined first, since it has a total dependency of zero. Once A is machined, the dependency values for surfaces/features B, C, D, and G are reduced to zero. Any of these surfaces/features could be selected. Face G is selected to be machined. This reduces the total dependency values for E



Material 1018 Steel
Standard Tolerances Apply When not Specified

Figure 4.3 Drill Sleeve Design After DFM Feasibility Analysis

Table 4.1 Dependency Matrix For Drill Sleeve

	A	B	C	D	E	F	G	Heat Treatment	Total
A	0	0	0	0	0	0	0	0	0
B	1	0	0	0	0	0	0	0	1
C	1	0	0	0	0	0	0	0	1
D	1	0	0	0	0	0	0	0	1
E	0	0	0	0	0	0	1	0	1
F	0	0	0	0	0	0	1	0	1
G	1	0	0	0	0	0	0	0	1
Heat treat & Finish	1	1	1	1	1	1	1	0	7

Table 4.2 Process Sequence For Drill Sleeve

Operation #	Surface Machined
10	Face A, Lap A
20	Face G, Lap G
30	Drill & Taper B
40	End Mill E End Mill F
50	Taper C
60	End Mill D
70	Heat treat & Finish

and F to zero. Now, B, C, D, E and F can be machined. An arbitrary sequence of B, E, F, C and D for the generation of surfaces/features is selected. Once all the surfaces/features are machined, dependency value for heat treatment becomes zero. Heat treatment and further finishing operations can be performed at this stage.

4.1.3 DFM Analysis For Drill Sleeve

In this section the process sequence generated is analyzed to determine the impact of tolerance stacking, relaxation of tolerances and configuration checking for the improvement of the process plan. The improvement of the process plan is achieved by eliminating unnecessary setups by combining operations. The diameters of the drill sleeve do not contribute to any tolerance stacking. The only dimensions that may cause any tolerance stacking is along the length of the sleeve. The analysis of the drill sleeve [Yue93] based on the process plan does not show any tolerance stacking.

In this part design the surface finish requirements on surface A and D are unnecessary, since they do not provide any function. If the surface finish requirements are eliminated, the lapping done on surface A and D can be avoided, thus reducing the number of operations.

The process plan sequence in Table 4.2 requires seven setups. The process plan has to be analyzed to determine if the number of setups can be reduced. A modified sequence based on the analysis is shown in Table 4.3. Since, surface A had a constraint of zero and all other surfaces/features have more than zero dependency value, surface A has to be produced first. This allows B, C, D and G to be produced. Since, machining A, B and C can be done using the same setup on turret lathe, machining B is the next operation selected.

Table 4.3 Alternate Process Sequence For Drill Sleeve

Operation #	Surface Machined
10	Face A Drill & Taper B Taper C
20	Face G
30	End Mill E End Mill F End Mill D
40	Heat treat & Finish

Surface C is machined next. Now D or G could be machined. But D, E and F requires milling operation which could be done in one setup. E and F require G to be done. Hence G is machined next. E, F and D can be milled in one setup now. Heat treatment and finish operations can be done next.

The modified process plan has been analyzed for tolerance stacking analysis and the results were found to be okay.

4.2 Case Study 2 - V-Block

In Case Study # 2, the manual DFM analysis of a V - block is performed. The initial part design submitted for manufacturability analysis is shown in Figure 4.4 and 4.5. The function of a V-block is to locate cylindrical parts, either in production or inspection as part of a jig or fixture. The V-block is usually 6 to 8 inches in length and the included angle of the V is usually 90 degrees. The design requirements for such a part are:

- Toughness, strength and hardness
- V has to be centrally located within +/- 0.0005 inch
- Parallelism within 0.00025 inch
- Squareness 0.0005 inch

The V-block has to be tough enough to withstand cutting forces, and other handling forces on the shop floor. Being a locating device, the V-block should not wear easily. The surface of the V-block has to be hard to prevent any wear on the V-Block. In addition, the surface of the V-block should be hard enough to preserve the parallelism and squareness tolerances, since it is used as a locating device. Given the high tolerances, toughness and hardness requirements, the V-block has to be hardened and ground.

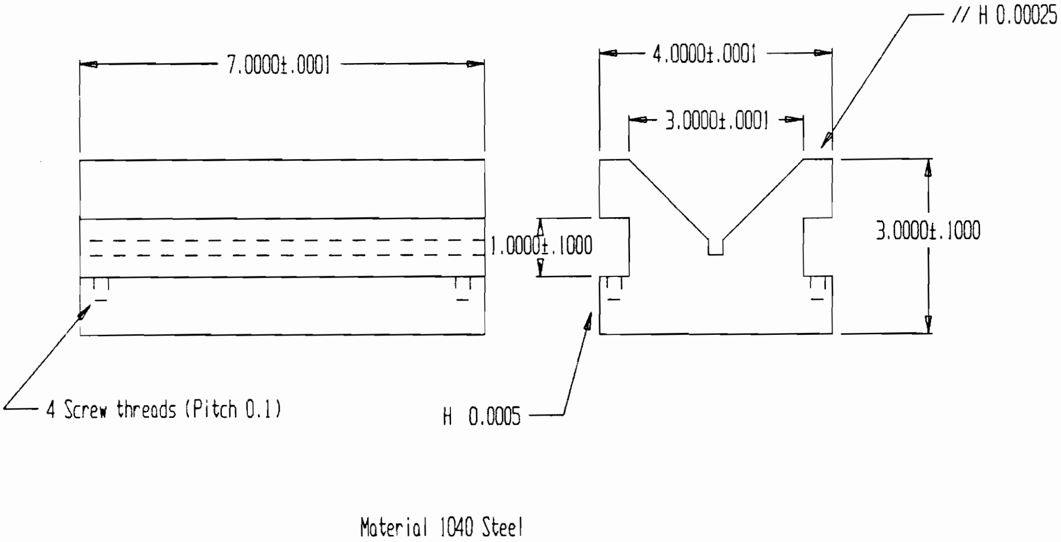


Figure 4.4 Initial V-Block Design with Dimensions

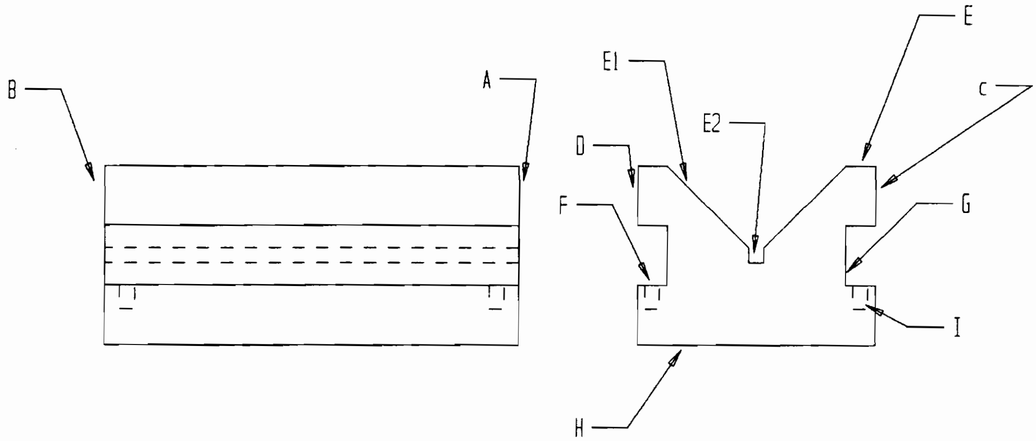


Figure 4.5 Initial V-Block Design with Labels

4.2.1 DFM Feasibility Analysis For the V-Block

The initial part design is subjected to a manufacturability analysis, prior to manufacturing.

Material analysis: The material selected by the designer is 1040 steel with no surface treatment specifications. Without any surface treatment the part does not have the wear resistance properties (hardness 160 Bhn). The toughness of 1040 steel is low. The material selected should have good hardenability as well. The machinability rating for 1040 steel is about 60. The material requirements for the V-block are very similar to that of the drill sleeve. Hence, 1018 steel which has higher manganese content and provides good machinability as well as hardenability properties is selected.

Dimensional checking: In the initial part design, the dimensions are not fully constrained. There is no reference surface provided. Important dimensions such as the angle of the V-block which are needed for the proper functioning of the V-block are missing. The dimensions were revised to minimize the reference surfaces and to avoid any possible mis-interpretations.

Conventional tolerance checking: The tolerances specified on linear dimensions are either too tight or too relaxed. If the tolerances do not serve any function, it is still important to provide the manufacturer with standard tolerances. Tolerances which are relaxed too much can also lead to manufacturing problems. If the V-block is to be manufactured using machining processes, the tolerances which are too tight has to be relaxed. The tolerances have been revised to reflect the values specified in the handbook.

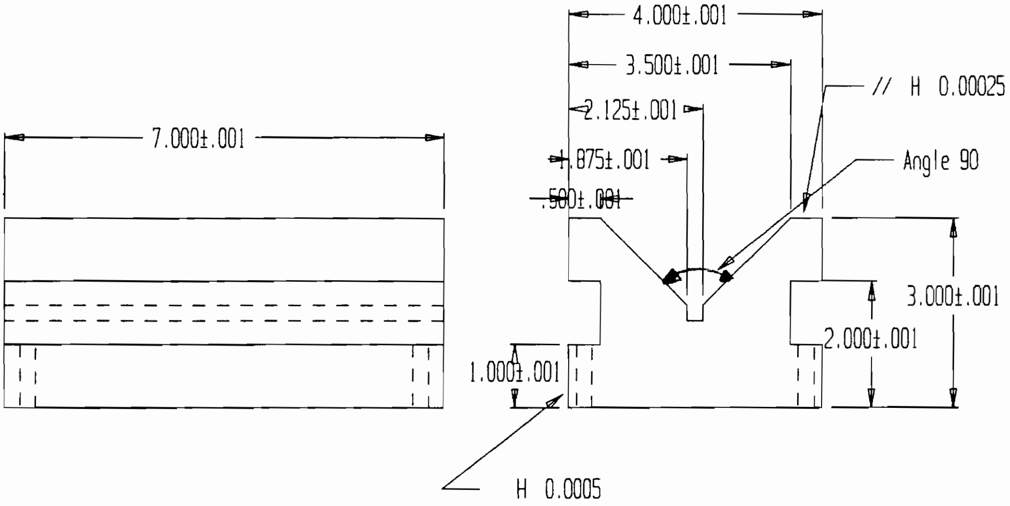
Geometric tolerance checking: The specifications of geometry tolerances are found to be within processing capabilities.

Configuration Checking: The V-block designed is to be assembled to a jig. The screw thread on the V-block was provided for assembly using a clamp and a nut. However, this screw thread cannot be machined since it is not accessible. By extending the screw thread, a configuration change which eliminates the clamp and allows the V-block to be assembled to the jig directly, will allow access to the screw thread to be machined. All the other surfaces that need to be machined are accessible to machine tools. It may be noted that by extending the screw thread the part can be manufactured. However, the assembly of the V-block to a jig will still pose problems, which will be revealed only in the design for assembly analysis. The part design after DFM feasibility analysis is shown in Figure 4.6.

4.2.2 Process Plan Sequencer for the V-Block

Using the process sequencer algorithm the dependency matrix table is first generated (Table 4.4). Surfaces A and H are the main reference surfaces. D and C has parallelism tolerance associated with surface H. C is also referenced with respect to D and hence total constraint for C is 2. E is referenced with respect to H, and hence has a constraint value of 1. The V-groove (E1), the thru_slot (E2), surface F and surface G are referenced from H, as well as D giving both a total constraint value of 2. Heat treatment is for all the surfaces and hence can be done only after all the features/surfaces have been generated (total constraint value of 10). The four thread_holes (I) cannot be tapped until heat treatment is completed and is referenced from A and from D (total constraint 3).

A process plan sequence based on this dependency table is given in Table 4.5. Since, the raw material is usually a long bar, cutting off to produce A is the first operation, which drops the total constraint for B to 0. Surfaces B and



Material : 1018 Steel

Figure 4.6 V-Block Design After DFM Feasibility Analysis

Table 4.4 Dependency Matrix For V-Block

	A	B	C	D	E	E1	E2	F	G	H	I	Heat Treatment	Total
A	0	0	0	0	0	0	0	0	0	0	0	0	0
B	1	0	0	0	0	0	0	0	0	0	0	0	1
C	0	0	0	1	0	0	0	0	0	1	0	0	2
D	0	0	0	0	0	0	0	0	0	1	0	0	1
E	0	0	0	0	0	0	0	0	0	1	0	0	1
E1 V-groove	0	0	0	1	0	0	0	0	0	1	0	0	2
E2 thru_slot	0	0	0	1	0	0	0	0	0	1	0	0	2
F thru_slot	0	0	0	1	0	0	0	0	0	1	0	0	2
G thru_slot	0	0	0	1	0	0	0	0	0	1	0	0	2
H	0	0	0	0	0	0	0	0	0	0	0	0	0
I - thread _hole	1	0	0	1	0	0	0	0	0	0	0	1	3
Heat treat & Finish	1	1	1	1	1	1	1	1	1	1	0	0	10

Table 4.5 Process Sequence For V-Block

Operation #	Surface Machined
10	Cutoff A
20	Mill B
30	Mill H
40	Mill D Mill F - thru_slot
50	Mill E Mill E2 - thru_slot Mill E1 - V-groove
60	Mill G - thru_slot Mill C
70	Heat treat & Finish
80	Tap I - thread_hole

H are produced next. E and D are the only surfaces with 0 constraint value. D was produced next. Once D is machined, C, E1, E2, F and G drop its total constraint values to 0. The thru_slot F is produced next. The surfaces E, E2 E1, G, and C were produced in sequence. Since all the surfaces/features have been generated, the heat treatment and finishing operations can be performed. Once the heat treatment operations are complete, the thread_holes can be tapped.

4.2.3 DFM Analysis For V-Block

The process sequence generated is analyzed to determine the impact of tolerance stacking, relaxation of tolerances and configuration checking for the improvement of the process plan. Tolerance stacking analysis does not show any problems. The process plan sequence in Table 4.5 requires eight setups. The process plan has to be analyzed to determine if the number of setups can be reduced or if the process sequence can be improved. A modified sequence based on the analysis is shown in Table 4.6. The alternate sequence generated requires the same number of setup as the initial sequence. However, the sequences has been changed to take into consideration the configuration of the part. For instance, operation # 40 had mill D and mill F. This setup has been changed to mill D and C in one setup, using a straddle mill. In operation 50 of the initial sequence, thru_slot (E2) is machined prior to the V-groove (E1). It would be much more easier to produce the V-groove and then produce the thru_slot.

In the next chapter the prototype software developed is described. The DFM analysis and results of three examples are also provided.

Table 4.6 Alternate Process Sequence For V-Block

Operation #	Surface Machined
10	Cutoff A
20	Mill B
30	Mill H
40	Mill D Mill C
50	Mill E Mill E1 - V-groove Mill E2 - thru_slot
60	Mill G - thru_slot Mill F - thru_slot
70	Heat treat & Finish
80	Tap I - thread_hole

CHAPTER 5

SOFTWARE PROTOTYPE

In this chapter, the implementation details of the prototype software are presented. This software is developed to demonstrate the DFM feasibility module, the process sequencing module of the DFM framework, and the data representation framework for the DFM analysis (Table 5.1). It does not perform some aspects of the manual methodology such as, material analysis, and DFM analysis. The software makes use of three examples to demonstrate the framework and the methodology. In Section 5.2, the model details and the results (DFM feasibility check) of the first example (E-Shaped part) is provided. In Section 5.3, the second example, which uses positive, negative and tertiary features, is used to demonstrate the process sequencer module. In Section 5.4, the third example (V-block), modeled using negative and tertiary features, is also used to demonstrate the process sequencer module.

5.1 Software Prototype

The developed system architecture is shown in Figure 5.1. The model data files contain feature, geometric, and topological information, as well as dimensions and tolerances information. The data representation follows the framework developed (see Section 3.2). Errors, such as, dimensional over-constraining, tolerance inconsistencies and infeasible part configurations are introduced in the models to demonstrate the analysis modules.

The DFM feasibility module consists of the following modules: a) Dimensional Checking, b) Conventional Tolerance Checking, c) Spatial Relationships, d) Geometrical Tolerance Checking, and e) Configuration Checking. The interface for the material analysis module in the sub menu has

Table 5.1 Software Prototype Specifications

DFM MODULE	DETAILS OF AUTOMATION
Dimensional Checking	Over dimensioning
Conventional tolerance Checking	Magnitude of tolerances
Generation of Spatial relationships	All spatial relationships defined are generated
Geometric tolerance checking	Check the accuracy's of specified tolerances
Configuration checking	Tool accessibility of all features defined
Process plan	Map surfaces/features to processes and generate sequence of machining

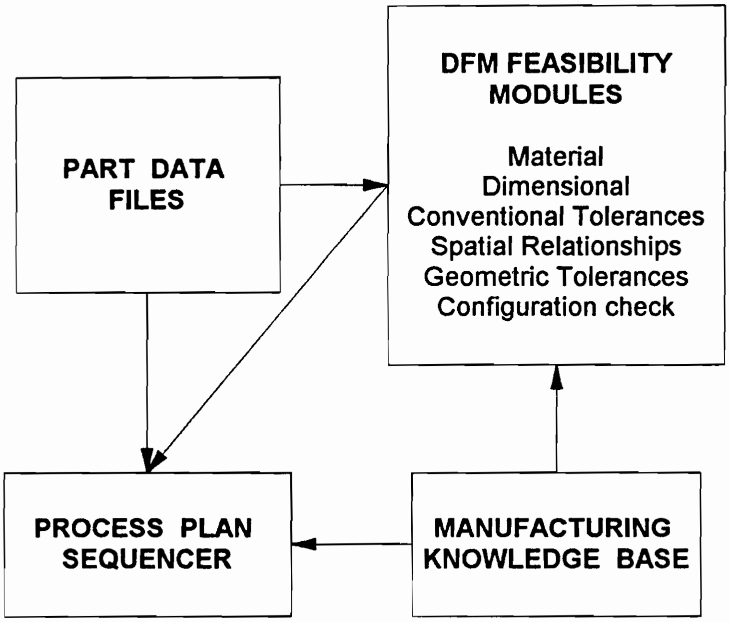


Figure 5.1 System Architecture

also been developed for future integration. The dimensional checking module determines whether the model is properly constrained with respect to the dimensions. The conventional tolerance check module determines whether the tolerances specified are too tight by performing general process capability checks. The spatial relationships module generates the spatial relationships between the features used in modeling the part. The geometrical tolerance check module determines tolerance inconsistencies (geometric tolerances) and general process capability checks. The configuration check module determines tool accessibility to all the features defined in the part. The user can activate any single module in the design feasibility module for analysis or may run the complete module.

The process plan module generates a feasible sequence for machining. The machining sequence generated is based on tolerance and dimensional constraints. It does not take into consideration the available machinery and its processing capabilities. The sequence generated is the input to the process plan generation module being developed by Chiang [Chia94]. The process plan generation module takes into consideration available manufacturing resources to check the sequence as well as to generate setups. Alternate sequences can also be generated using this module. This could lead to a true generative CAPP system.

The manufacturing knowledge base contains information about the general processing capabilities for milling, turning, and drilling. The information is stored in an object oriented manner in the process_capability class. This class is further derived to the various processing classes, such as milling, drilling etc. The processing capabilities for linear dimensions, surface finish, perpendicularity, parallelism etc. are provided in this class. The manufacturing knowledge base is being extended by Chiang [Chia94] to include specific instances of available machinery, which can be used to represent individual factory setups.

The user interface for the software is shown in Figure 5.2. The user interface allows different model data files to be selected. There are four model data files that can be selected. The sub-modules in the DFM feasibility analysis module allows each of the modules to be selected for analysis. Selecting the process plan module will activate the process sequencer module.

The software flow diagram is shown in Figure 5.3. Upon activating the software, the user selects a model for analysis. Once the model is loaded, the user can perform a DFM feasibility analysis or generate the process sequence. In the DFM feasibility analysis, the model can either run each module for analysis or perform a sequential analysis. After the DFM feasibility analysis, the user can load a new model, generate process sequence, or exit out of the program. After the process sequence is modeled the user can load a new model or exit out of the program.

5.2 Example 1 - E-Shaped Part

The first case study shown in Figure 5.4 and 5.5 is used to demonstrate the DFM feasibility module (dimensional check, spatial relationship, tolerance check and part configuration analysis module). The E-shaped part design is often cited in literature to illustrate the need for spatial relationships to perform tool accessibility analysis. Various errors have been introduced into the part design to demonstrate each of the modules used for DFM analysis. The design has been overdimensioned and the tolerances are too tight. The geometrical tolerances applied are not consistent with the labeling of the surfaces. The configuration of the part is also such that the hole_A is not accessible.

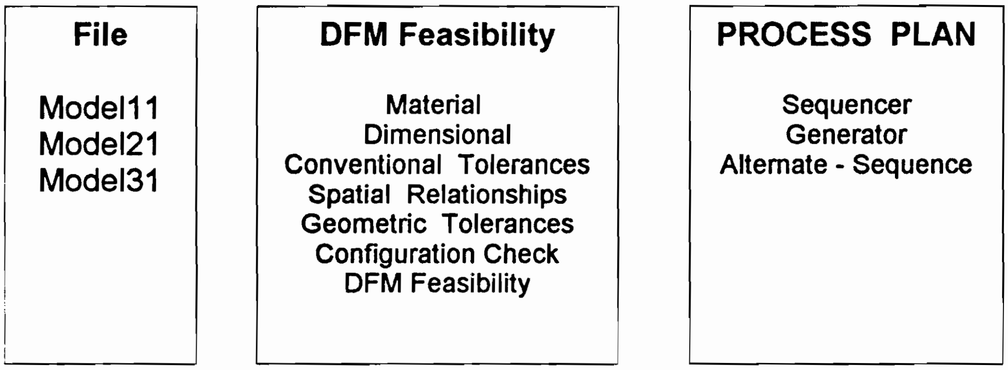


Figure 5.2 User Menu System

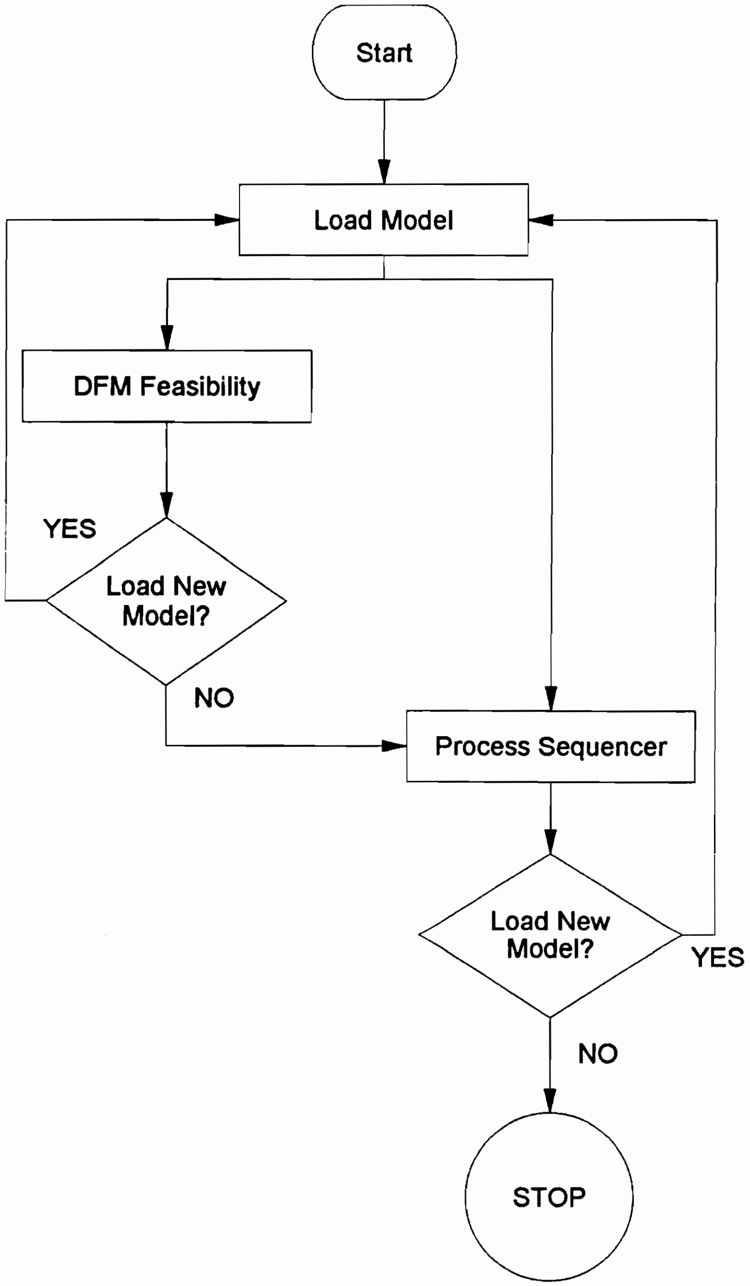


Figure 5.3 Software Flow Diagram

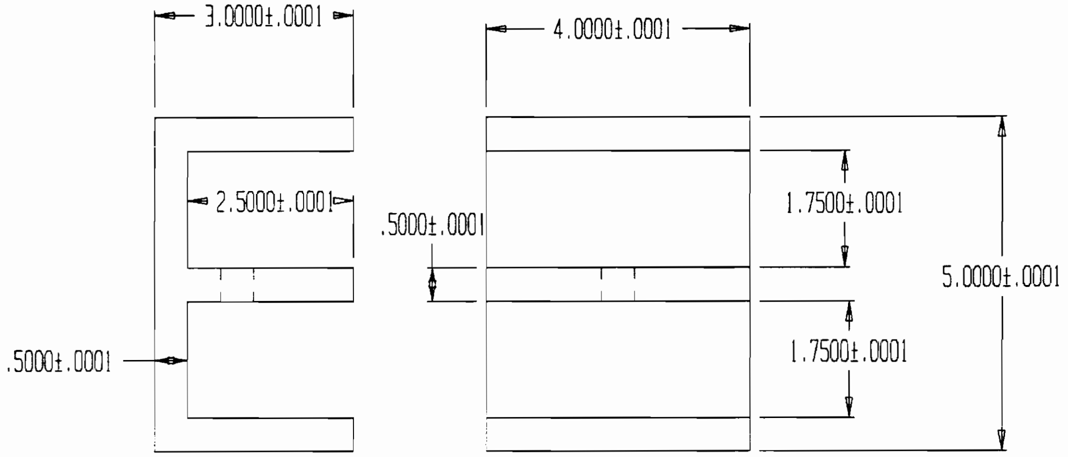


Figure 5.4 Dimensioned E-Shaped Part

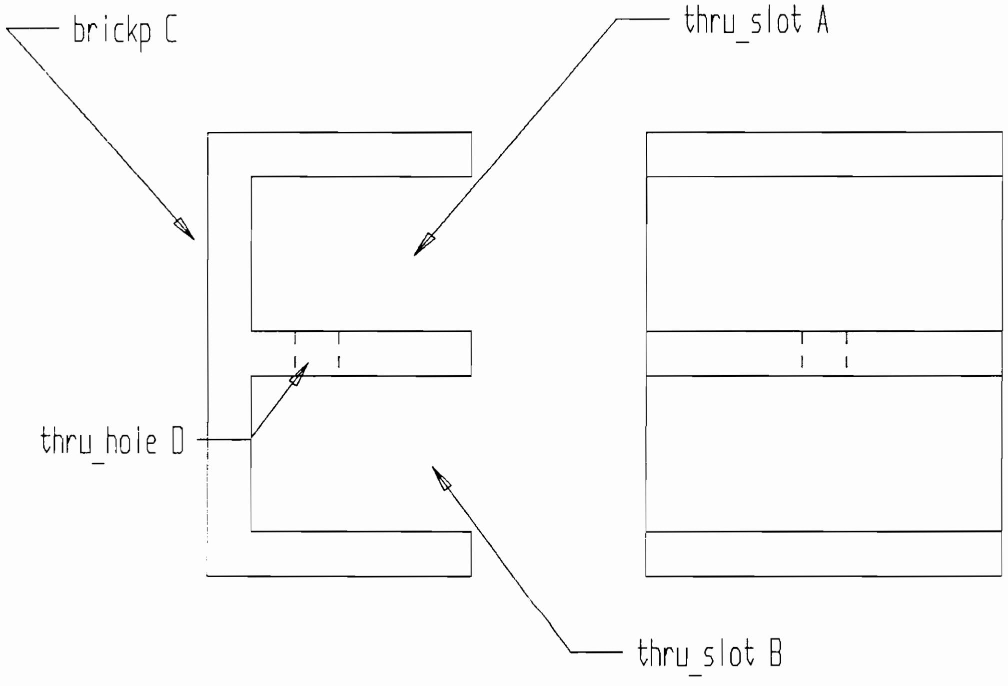


Figure 5.5 Labeled E-Shaped Part

The DFM feasibility module was used to analyze the design. The results of the analysis are written to an output file as shown in Table 5.2. The dimensional checking identifies all the dimensions along the X and Z- axis (where it was overdimensioned). All the conventional tolerance specifications are too tight. The configuration module output based on the spatial relationships has determined that there is no tool accessibility for thru_hole D.

5.3 Example 2 - Round Part

The second example is shown in Figure 5.6 and 5.7. This example is used to demonstrate the DFM process sequencing module. Any CAD model can be modeled using positive features, negative features or a combination of both (worst case). The algorithms used for sequencing have to take into account the possibility of each of these cases. The part design for this case study is modeled using positive, negative and tertiary features. The three pieces of the shaft are modeled as positive features. The groove and the cylindrical hole are modeled as negative features. The chamfers are modeled as tertiary features. Features such as heat treatment, chamfer, etc. can be modeled as tertiary features. Tertiary features are associated to the surfaces on which they are to be generated. The number of surfaces that the tertiary feature is associated with, will determine the no of constraints. A process sequence has been generated automatically (Table 5.3) for this part based on the dimensioning and tolerance requirements. Surface J is machined first since, this surface is the datum for all other surfaces. Surface E and H can be turned next. Chamfer I and groove g are defined on surface E and H and hence can be turned next. The thru_hole F does not depend on any surface and is produced next. Surface A is faced next, C followed by turning C, B and D. This sequence though not the optimum one is valid and does not violate any precedence requirements.

Table 5.2 DFM Feasibility Analysis Output for E-Shaped Part

Dimensional check output	
Over dimensioning along:	Z-Axis
Check the following:	5.0000 1.7500 1.7500 0.5000
Over dimensioning along:	X-Axis
Check the Following:	3.0000 2.5000 0.5000

Conventional tolerances output	
Too tight tolerances:	Z-Axis
Check the following:	5.0000 1.7500 1.7500 0.5000 4.0000 3.0000 2.5000 0.5000

Configuration analysis output
Thru_hole D is not accessible

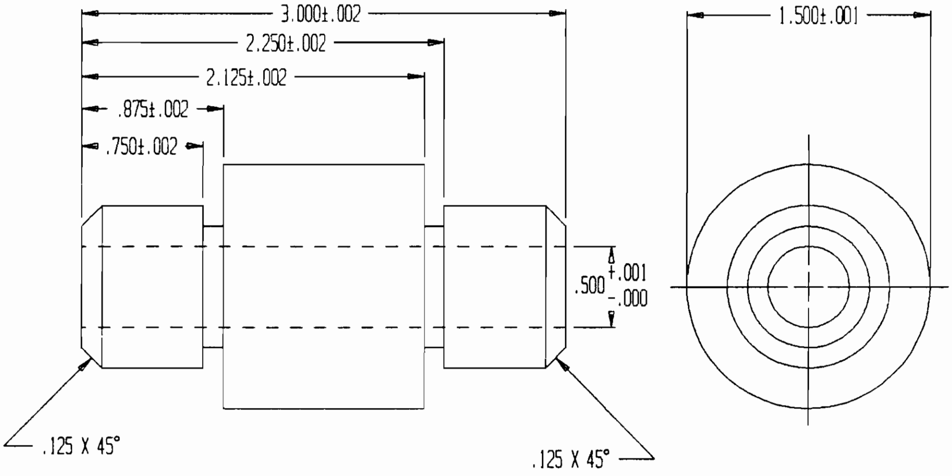


Figure 5.6 Dimensioned Round Part

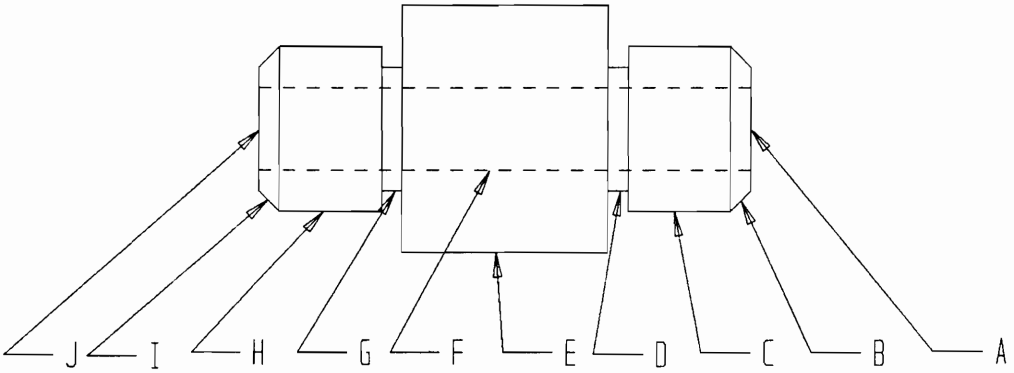


Figure 5.7 Labeled Round Part

Table 5.3 Process Sequence For Round Part

Sequence #	Surface To Be Machined
1	Machine J
2	Turn E
3	Turn H
4	Chamfer I
5	Cut Groove G
6	Drill F
7	Machine A
8	Turn C
9	Chamfer B
10	Cut Groove D

5.4 Example 3 - V-Block

Figure 5.8 and 5.9 shows the V-block design which is also used to demonstrate the DFM process sequencing module. This part is modeled using negative and tertiary features. The raw material is specified as a positive brick feature. The thru_slots and V-groove are specified as negative secondary features. Heat treatment is specified as a tertiary feature. A process sequence has been generated (Table 5.4) for this part based on the dimensioning and tolerance requirements. Since surface A is the datum, it is milled first, followed by surface B. Due to the parallelism and perpendicularity constraints on E and D with respect to the bottom surface H, H is milled next. Surface D and thrus_slot F are machined next. Surface E has to be machined prior to machining features E2 and E1. Surface C and thru_slot G are machined next. This sequence is valid and does not violate any precedence requirements.

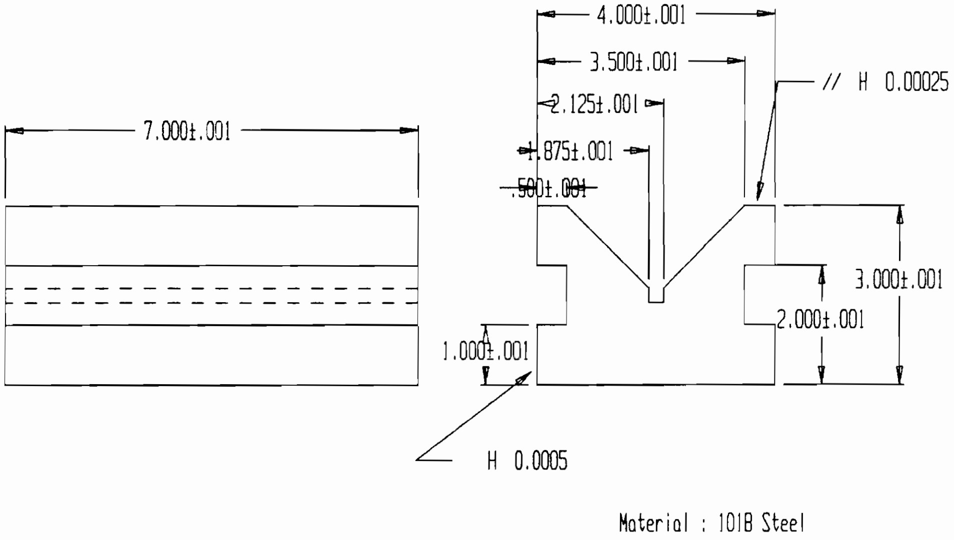


Figure 5.8 Dimensioned V-Block

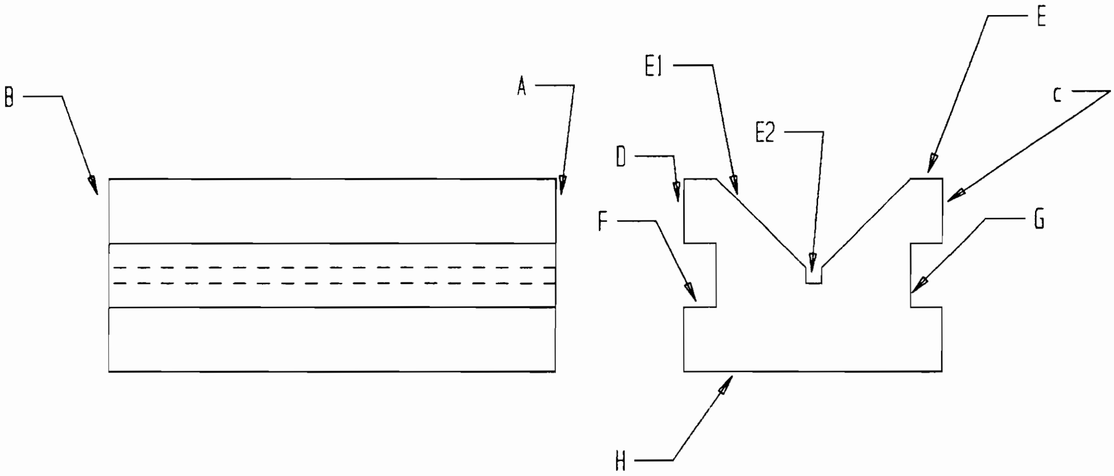


Figure 5.9 Labeled V-Block

Table 5.4 Process Sequence For V-Block

Sequence #	Surface To Be Machined
1	Mill A
2	Mill B
3	Mill H
4	Mill D
5	Mill F
6	Mill E
7	Mill E2
8	Mill E1
9	Mill G
10	Mill C
11	Heat treat

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

6.1 Summary and Conclusions

Integration of manufacturing considerations into the product design phase is necessary for the effective automation of design for manufacturability (DFM) analysis. The sequential process of decision making in traditional product development process results in increased costs and higher product development times. With the trend towards better product quality, product customization, shorter product life cycle, and international competition, manufacturers are faced with the challenge of improving product quality while reducing product time, manufacturing lead-time and product cost. To cope with these challenges, the product development process has to be made more efficient by integrating manufacturing and assembly considerations in the design phase itself, through the use of techniques such as (DFM) and Design For Assembly (DFA).

DFM techniques have to be automated to take advantage of the vast advances in CAD and CAM systems. However, the automation of DFM has been constrained, especially for machined components, by the lack of methodologies, which are dependent on the process of manufacture, and the incomplete part data representation in CAD systems. A search of the literature indicates only two methodologies: one in welding and the other in sheet metal work. A methodology for DFM analysis of machined products did not exist. This research has developed a DFM methodology for machined components.

The DFM methodology consists of three modules: DFM feasibility, process plan generation and DFM analysis. The DFM feasibility module performs an initial feasibility check on the material, dimensions, tolerances, and configuration of the part. It also generates the spatial relationships between features. The process plan generation module uses a sequence identifier algorithm to generate the manufacturing sequence based on design specifications. The DFM analysis module evaluates tolerances relative to their stacking effects and manufacturability, and analyzes the part configuration for possible design and process plan improvements. The DFM methodology has been validated using two case studies. The first case study uses a drill sleeve and the second case study makes use of a V-block design to demonstrate the methodology.

In order to automate the developed methodology, the data required for analysis as well its representation framework has to be identified. The data representation in currently available CAD systems do not address the issue of representation of manufacturing data, such as spatial relationships between features and tool accessibility of feature. In addition, machining requires complete geometric data of features to be represented and retained. This research identified the data required for analysis and developed an appropriate data representation framework which addresses the special requirements of manufacturability analysis for machined components. The processing capabilities of machining processes are also addressed in the data representation framework. In addition, a software prototype has been developed to address the issues of data representation.

In the software prototype, dimensions checking, tolerances checking, configuration checking and spatial relationships generation in the DFM feasibility module are automated. In addition the process sequencer module is also automated. Three examples were used to demonstrate the software. The first example uses an E-shaped object with infeasible part configuration, tight tolerances and erroneous representation of geometric tolerances to demonstrate the automated DFM feasibility modules. The second example uses a round shaped part modeled using positive and negative features (worst case scenario) to demonstrate the process sequencer module. The V-block used to describe the manual methodology is used as the third example and is used to demonstrate the process sequencer module as well. The automated process sequence generation module provides the sequence of surfaces/features to be machined. The software is developed on an IBM compatible machine using C++.

The first major contribution of this work is the development of a DFM methodology that can be automated for machined components. The second major contribution of this work is the identification of data required for DFM analysis and its data representation. This representation would contribute to enhancing the PDES/STEP standards. The representation of chamfer, heat treatment etc. as tertiary features is a new concept and is required for process planning. The sequencer algorithm for process planning could lead to the development of a true regenerative CAPP system, which takes into consideration the influence of tolerances specified on machine selection and process identification. The representation of the process plan as entities will aid in the complete automation of the DFM analysis process.

6.2 Recommendations for Future Work

In this research the part data was read from data files. The software has to be integrated with a feature-based CAD modeler, such as Pro-Engineer. This is necessary for easy modeling and to perform part design changes as analysis is performed.

The issue of representation of manufacturing resources has to be addressed. Standard formats for the representation of machinery, tools, and process capability have to be addressed. This is being addressed by Chiang [Chia94] as part of his M.S. Thesis.

Various modules of the DFM methodology which are not automated in the current software, such as, material selection, tolerance control and assembly analyses has to be developed. This research utilizes a sequential analysis which could lead to an infinite loop. The sequential nature of analysis can be eliminated by a blackboard architecture control of the software. The research is focused on a few processes in metal removal, namely turning, milling and drilling. The research and software has to be extended to other machining processes - shaper, finishing operations (grinding, lapping etc.), and non-traditional methods of metal removal processes. An elaborate research to determine a comprehensive set of rules for manufacturability analysis has to be undertaken. In addition, the DFM methodology and concept has to be extended to cover other methods of manufacturing, such as casting, forming processes, joining processes and non-traditional processes.

References

[Bloo89] Bloom, H.M., "Design for Manufacturing and the Life Cycle," Design Theory '88, (New York: Springer-Verlag), 1989.

[Boot83a] Boothroyd, G., and Dewhurst, P., "Design for Assembly: Manual Assembly," Machine Design, December 1983, pp. 140 - 145.

[Boot83b] Boothroyd, G., and Dewhurst, P., "Design for Assembly: Selecting the Right Method," Machine Design, November 1983, pp. 94 - 98.

[Chan84] Chang, T.C., and Wysk, R.A., "Integrating CAD and CAM Through Automated Process Planning," International Journal of Production Research, Vol. 22, No. 5, 1984, pp. 877-894.

[Chan88a] Chang, T.C., et al., "QTC - An Integrated Design/Manufacturing/Inspection System for Prismatic Parts," *Proceedings of the ASME Computers in Engineering Conference*, SF, Cal., vol. 1, 1988, pp. 417 - 426.

[Chan89] Chang, S.S.L., "A Fuzzy Set Algorithm for Engineering Design Applications to the Component Parts Industry," Design Theory '88, (New York: Springer-Verlag), 1989.

[Chan91] Chang, T.C., and Joneja, A., "Search Anatomy in Feature-Based Process Planning," Journal of Design and Manufacturing, vol. 1, 1991, pp. 7 - 15.

[Chan90] Chang, T.C., *Expert Process Planning Systems*, Addison-Wesley, Reading, Massachusetts, 1990.

[Chen89] Chen, C.S., "Developing A Feature Based Knowledge System for CAD/CAM Integration," Computers in Industrial Engineering, vol. 15 - 1, 1988, pp. 34 - 40.

[Chen91a] Chen, Y.M., Miller, R.A., and Vemuri, K.R., "On Implementing an Integrated Manufacturability Assessment Environment," Computers in Engineering, 1991, pp. 407 - 413.

[Chen91b] Chen, Y.M., Miller, R.A., and Vemuri, K.R., "A Framework for Feature Based Part Modeling," Computers in Engineering, 1991, pp. 357 - 365.

[Chen92] Chen, Y.C., Miller, R.A., and Lu, S.C., "Spatial Reasoning on Form Feature Interactions for Manufacturability Assessment," Computers in Engineering, vol. 1, 1992, pp. 29 - 36.

[Chen94] Chen, C.L., and LeClair, S.R., “Integration of Design and Manufacturing: Solving Setup Generation and Feature Sequencing Using an Unsupervised-Learning+,” Computer Aided Design, vol. 26, No. 1, 1994, pp. 59 - 75.

[Chia94] Chiang, C.Y., *Process Planning for Machined Components Using A Sequencer Algorithm,* Unpublished Masters Thesis, Virginia Polytechnic Institute and State University, December 1994.

[Chua89] Chuang, S.H., and Henderson, M.R., "Three Dimensional Shape Pattern Recognition using Vertex Classification and Vertex-edge Graphs", Computer-Aided Design, July 1989, pp. 379-385.

[Corb91] Corbett, J., et al., *Design For Manufacture - Strategic Principles and Techmiques*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1991.

[Cutk89] Cutkosky, M.R., Brown, D.R., and Tenenbaum, J,M., Extending Concurrent Product and Process Design Towards Earlier Design Stages," Concurrent Product and Process Design - DE, vol. 21, 1989, pp. 65 - 72.

[Cutk90] Cutkosky, M.R., and Tenenbaum, J.M., "Methodology and Computational Framework for Concurrent Product and Process Design," Mechanism & Machine Theory, vol. 25, no. 3, 1990, pp. 365 - 381.

[Dasi90] Da Silva, R.E., Wood, K.L., and Beaman, J.J., "Representing and Manipulating Interacting and Interfeature Relationships in Engineering Design for Manufacture," Advances in Design Automation - DE , vol. 23 - 1, 1990, pp. 1 - 8.

[Deis92] Deisenroth, M.P., Terpenny, J.P. and Eyada, O.K., "Design for Competitiveness: Case Studies Developed in Concurrent Engineering and Design for Manufacturability," *Proceedings, ASME Winter Annual Meeting*, DE-Vol. 51, Anaheim, California, November 1992, pp. 29-34.

[Dixi89] Dixit, S., *Expert System for Machinability Data Integrated with a CAD System,* Unpublished Masters Thesis, Virginia Polytechnic Institute and State University, August 1989.

[Dixo88a] Dixon, J.R., "Design with Features: Building Manufacturing Knowledge into More Intelligent CAD Systems," Manufacturing International, vol. 1, 1988, pp. 51 - 57.

[Dixo88b] Dixon, J.R., "On Research Methodology Towards a ScientificTheory of Engineering Design," Design Theory '88, (New York: Springer-Verlag), 1989.

[Dym91] Dym, C.L., and Levitt, R.E., *Knowledge Based Systems in Engineering*, New York, McGraw-Hill, 1991.

[Eyad93a] Eyada, O.K., Christein, J.P., and Krishnan, K.K., "A DFM Framework for Integrating Arc Welding Parameters into Design," Transactions, NAMRI/SME, vol. XXI, 1993, pp. 375 - 382.

[Eyad93b] Eyada, O.K., and Krishnan, K.K., "Assessment of Feature Recognition Algorithms in Solid Modeling Representation," *Proceedings, Third International Flexible Automation and Integrated Manufacturing,* Limerick, Ireland, June 1993, pp. 232 - 244.

[Fabr91] Fabrycky, W.J., *Life-Cycle Cost and Economic Analysis*, Prentice Hall, Englewood Cliffs, NJ, 1991.

[Falc87] Falcidieno, B., and Giannini F., "Extraction and Organization of Form Features into a Structured Boundary Model," Eurographics Association 1987, pp. 249-259.

[Fing89] Finger, S., and Dixon, J.R., "A Review of Research in Mechanical Design. part ii:," Research in Engineering Design, vol. 1, No. 2, 1989, pp. 127 - 137.

[Flor87] Floriani, L.D., "A Graph-Based Approach to Object-Oriented Feature Recognition," ACM, 1987.

[Flor89] Floriani, L.D., and Bruzzone, E., "Building a Feature Based Object Description from a Boundary Model," Computer Aided Design, vol. 21, no. 10, 1989, pp. 602 - 609.

[Fu82] Fu, K.S., *Syntactic Pattern Recognition and Applications*, New Jersey, Prentice-Hall, 1982.

[Gand89] Gandhi, A., and Myklebust, A., "A Natural Language Approach to Feature Based Modeling," Advances in Design Automation - DE, vol. 19 - 1, 1989, pp. 69 - 77.

[Gava90] Gavankar, P., and Henderson, M.R., "Graph Based Extraction of Protrusions and Depressions from Boundary Representations," Computer-Aided design, Vol. 22 - 7, 1990, pp. 442-450.

[Haye89] Hayes, C.C., Desa, S., and Wright, P.K., "Using Process Planning Knowledge to Make Design Suggestions Concurrently," Concurrent Product and Process Design - DE, vol. 21, 1989, pp. 87 - 92.

[Hend84] Henderson, M.R., and Anderson, D.C., "Computer Recognition and Extraction of Form Features : A CAD/CAM Link," Computers in Industry, No. 5, 1984, pp. 324-339.

[Hend88] Henderson, M.R., and Chang, G.J., "FRAPP: Automated Feature Recognition and Process Planning from Solid Model Data," American Society of Mechanical Engineers.

[Hodg91] Hodgson, B.A., and Pitts, G., "Designing for CNC Manufacture," Design For Manufacture - Strategic principles and techniques, (England: Addison-Wesley) (1991), pp. 56-68.

[Hoth90] Hothwaite, B., "Checklist for DFM," Machine Design, February 1990, pp. 163 - 167.

[Iran89] Irani, R.K., Kim, B.H., and Dixon, J.R., "Integrating CAE, Features, and Iterative Redesign to Automate the Design of Injection Molds," Computers in Engineering, vol. 1, 1989, pp. 27 - 33.

[Humm89a] Hummel, K.E., "Coupling Rule-Based and Object-Oriented Programming for the Classification of Machined Features," Computers in Engineering, 1989, pp. 409 - 418.

[Humm89b] Hummel, K.E., and Brown, C.W., "The role of features in the Implementation of Concurrent Product and Process Design," Concurrent Product and Process Design - DE, vol. 21, 1989, pp. 1 - 8.

[Ishi90] Ishii, K., Lee, C.H., and Miller, R.A., "Methods for Process Selection in Design," Design Theory and Methodology - DE, vol. 27, 1990, pp. 105 - 112.

[Ishi92a] Ishii, K., and Miller, R.A., "Design Representation for Manufacturability Evaluation in CAD: Beyond Feature Based Design," Computers in Engineering, vol. 1, 1992, pp. 37 - 44.

[Ishi92b] Ishii, K., and Hornberger, L., "The Effective Use and Implementation of Computer-Aids for Life-Cycle Product Design," Advances in Design Automation - DE , vol. 44 - 1, 1992, pp. 367 - 374.

[Jone91] Joneja, A., and Chang T.C, "Search Anatomy in Feature-Based Automated Process Planning," Journal of Design and Manufacturing, vol. 1, 1991, pp. 7 - 15.

[Josh88] Joshi, S., and Chang, T.C., "Graph-Based Heuristics for Recognition of Machined Features from a 3D Solid-Model," Computer-Aided Design, March 1988, Vol. 20, No. 2, pp. 58-60.

[Kim88] Kim, S.H., "A General Model of Design: Formalization and Consequences," Manufacturing Review, 1, 1988, pp. 109-116.

[Laak90] Laakko, T., et. al., "Feature Models for Design and Manufacturing," Proceedings of the Hawaii International Conference on System Science, vol. 2, 1990, pp. 445 - 454.

[Lee87] Lee, Y.C., and Fu K.S., "Machine Understanding of CSG: Extraction and Unification of Manufacturing Features," IEEE Computer Graphics and Applications, vol. 7, no. 1, 1987, pp. 20 - 32.

[Li88] Li, R.K., and Bedworth, D.D., "A Framework for the Integration of Computer-Aided Design and Computer-Aided Process Planning," Computers and Industrial Engineering, Vol. 14, No. 4, 1988, pp. 395-413.

[Li91] Li, R.K., Taur, B.W., and Shyur, H.J., "A Two-Stage Feature-Based Design System," International Journal of Production Research, vol. 29 - 1, 1991, pp. 133 -154.

[Lin92] Lin, W., *An Object-Oriented Software Development Environment For Geometric Modeling in Intelligent Computer Aided Design*, Unpublished Doctoral Dissertation, Virginia Polytechnic Institute and State University, 1992.

[Liu91] Liu, H., and Nnaji, B.O., "Design with Spatial Relationships," Journal of Manufacturing Systems, vol. 10, no. 6, 1991, pp. 449 -

463.

[Lu88] Lu, S., and Subramanyam, S., "A Computer-Based Environment for Simultaneous Product and Process Design, Proceedings of the ASME Winter Annual Meeting, vol. 31, New York, ASME Production Engineering Division, 1988, pp. 35-46.

[Padh90] Padhy, S.K., Sharan, R., and Dwivedi, S.N., "Feature Based Approach for Casting Design," Design Theory and Methodology - DE, vol. 27, 1990, pp. 113 - 118.

[Pern89] Perng, D.B., Chen, Z., and Li, R.K., "Automatic 3D Machining Feature Extraction from 3D CSG Solid Input," Computer-Aided Design, September 1989, pp. 285-294.

[Rind89] Rinderle, J.R., et al., "Form-Function Characteristics of Electro-Mechanical Designs," Design Theory '88, (New York: Springer-Verlag), 1989.

[Rose92a] Rosen, D.W. et al., "Features and Algorithms for Tooling Cost Evaluation in Injection Molding and Die Casting," Computers in Engineering, vol. 1, 1992, pp. 45 - 52.

[Rose92b] Rosen, D.W., and Grosse, I.R., "A Feature Based Shape Optimization Technique for the Configuration and Parametric Design of Flat Plates," Engineering with Computers, vol. 8, 1992, pp. 81 - 91.

[Schm89a] Schmitz, J., and Desa, S., "The Devlopment and Implementation of a Design for Producibility Method for Precision Planar Products," *Proceedings of the 15th ASME Automation Conference*, Canada, 1989, pp. 17 - 21.

[Schm89b] Schmitz, J., and Desa, S., "The Application of a Design for Producibility Methodology to Complex Stamped Products," Concurrent Product and Process Design, DE. - vol. 21, 1989, pp.

169 - 174.

[Shah88] Shah, J.J., and Rogers, M.T., "Functional Requirements and Conceptual Design of the Feature-Based Modeling System," Computer Aided Engineering Journal, 1988, pp. 9 - 15.

[Stal83] Staley, S.M., Henderson, M.R., and Anderson, D.C., "Using Syntactic Pattern Recognition to Extract Feature Information from a Solid Geometric Database," Computers in Mechanical Engineering, Vol. 2, No. 2, September 1983, pp. 61-66.

[Stol89] Stoll, H., and Hothwaite, B., "DFA Promises and Delivers," edited by Schuch, L.K., Assembly Engineering, 1989, pp. 18 - 21.

[Stol91] Stoll, W.H., "Design For Manufacture: An Overview," Design For Manufacture - Strategies Principles and Techniques (Addison-Wesley Publishing Company), 1991, pp. 107 - 129.

[Tons94] Tonshoff, H.K., Baum, T., and Ehrmann, M., "SESAME: A System for Simultaneous Engineering," Proceedings of the Fourth International FAIM Conference, 1994, pp. 380 - 389.

[Winn88] Winner, R.I., Pennel, J.P., Bertrand, H.E., and Slusarczuk, M.M.G., "The Role of Concurrent Engineering in Weapon Systems Acquisition," IDA Report R - 338, 1988.

[Yu92] Yu, T.T., et al., "A Framework for Sheet Metal Part and Manufacturing Assessment," Computers in Engineering, vol. 1, 1992, pp. 53 - 60.

[Yue93] Yue, J., *A Computerized Optimization Method for Tolerance Control,* Unpublished Masters Thesis, Virginia Polytechnic Institute and State University, November 1993.

Bibliography

[Agog87] Agogino, A.M., and Rege, A., "IDES: Influence Diagram Based Expert System," Mathematical Modeling, 8, 1987, pp. 227-233.

[Agog89] Agogino, A.M., Imanio, W., and Wang, S.S., "Decision-Analytical Method for Life Cycle Design," Design Theory '88, (New York: Springer-Verlag), 1989.

[Ahma88] Ahmad, K., and Macdonald, A., "An Expert System on Tolerances in Mechanical Design," *Proceedings of the ESD/SMI Expert Systems Conference and Exposition for Advanced Manufacturing Technology*, Dearborn, Michigan, pp. 309-326.

[Alde88] Aldefeld, B., "Variation of Geometries Based on A Geometric-Reasoning Method," Computer Aided Design, vol. 20, no. 3, 1988, pp. 117 - 126.

[Ande90] Anderson, D.C., and Chang T.C., "Geometric Reasoning in Feature-Based Design and Process Planning," Computers and Graphics, vol. 14, 1990, pp. 225 - 235.

[Banc92] Bancroft C.E., et al., Bakerjian R., (eds.): *Tool and Manufacturing Engineers Handbook*, vol. 6, ed. 4. Dearborn, Society of Manufacturing, 1992, chap 10, pp. 1-40.

[Bao88] Bao, H.P., "An Expert System for SMT Printed Circuit Board Design for Assembly," Manufacturing Review, vol. 1, no. 4, 1988, pp. 87 - 96.

[Bard92] Bardsz, T., and Zeid, I., "A Blackboard Architecture to Support Case-Based Reasoning in Mechanical Design," Advances in Design Automation - DE , vol. 44 - 1, 1992, pp. 337 - 343.

[Bell90] Bell, D.G., and Taylor, D.L., "Determining Optimal Model Complexity in an Iterative Design Process," *Proceedings of 2nd*

International Conference on Design Theory and Methodology, vol. 27, Chicago, ASME Design Engineering Division, Sep. 1990, pp. 291 - 297.

[Bill89] Billo, R.E., Henderson, M., and Rucker, R., "Applying Conceptual Graph Inferencing to Feature-Based Engineering Analysis," Computers in Industry, vol. 13, 1989, pp. 195 -214.

[Bour89] Bourne, D., Navinchandra, D., and Ramaswamy, R., "Tolerance-Free Designs for Manufacturing," Concurrent Product and Process Design - DE, vol. 21, 1989, pp. 47 - 56.

[Bral86] Bralla, J.G., (ed.), *Handbook of Product Design for Manufacturing*, McGraw-Hill, Inc., USA, 1986.

[Burn90] Burns, S.A., "Generic Classes of Iterative Behavior in Engineering Design Optimization," Design Theory and Methodology - DE, vol. 27, 1990, pp. 299 - 305.

[Chan85] Chang, T.C., and Wysk, R.A., *An Introduction to Automated Process Planning Systems*, Prentice Hall, Englewood, New Jersey, 1985.

[Chan93] Chan, K.C., and Nhieu, J.A., "A Framework for Feature-Based Applications," Computers in Engineering, vol. 24, NO. 2, 1993, pp. 151 - 164.

[Chen90] Chen, A., et al., "Design History Knowledge Representation and Its Basic Computer Implementation," Design Theory and Methodology - DE, vol. 27, 1990, pp. 175 - 184.

[Chov90] Chovan, J.D., and Waldron, M.B., "The Use of Function and Form When Reading Mechanical Engineering Drawings by Experts and Non-Experts," Design Theory and Methodology - DE, vol. 27, 1990, pp. 137 - 144.

[Chun90] Chung, J.C.H., and Schussel, M.D., "Technical Evaluation of

Variational and Parametric Design," Computer in Engineering, vol. 1, 1990, pp. 289 - 297.

[Cobb84] Cobb, M.F., "Design for Assembly is the Key to Increased Productivity," RCA Engineer, 29, 1984, pp. 4-10.

[Dewh84a] Dewhurst P., and Boothroyd, G., "Design for Assembly: Automatic Assembly," Machine Design, January, 1984, pp. 87 - 92.

[Dewh84b] Dewhurst P., and Boothroyd, G., "Design for Assembly: Robots," Machine Design, February, 1984, pp. 72 - 76.

[Dong90] Dong, X., and Wozny, M., "Managing Feature Type Dependency in a Feature Based Modeling System," Computers in Engineering, vol. 1, 1990, pp. 125 - 130.

[Eary62] Eary, D.F., and Johnson, G.E., *Process Engineering for Manufacturing*, Prentice-Hall, Inc. Englewood Cliffs, NJ, 1962.

[Elma91] ElMaraghy, H.A., et al., "Design for Assembly of a Direct Current Motor," Design For Manufacture - Strategic Principles and Techniques, (England: Addison-Wesley), 1991, pp. 174 - 187.

[Ferr89] Ferreira, J.C.E., and Hinduja, S., "Convex Hull-Based Feature-Recognition Method for 2.5D Components," Computer-Aided Design, July 1989, pp. 41-49.

[Fing90] Finger, S., and Safier, S.A., "Representing and Recognizing Features in Mechanical Designs," Design Theory and Methodology - DE, vol. 27, 1990, pp. 19 - 25.

[Fitz81] Fitzgerald, W.J., "Using Axial Dimensions to Determine the Proportions of Line Drawings in Computer Graphics," Computer Aided Design, vol. 13, no. 6, 1981, pp. 377 - 382.

[Fogl90] Fogle, M.A., and Wu, J.K., "A Relative Coordinate Formulation for Variational Solid Modeling," Design Theory and Methodology - DE, vol. 27, 1990, pp. 11 - 18.

[Gadh89] Gadh, R., Hall M.A., Gursoz, E.L., and Prinz, F.B., "Knowledge Driven Manufacturability Analysis from Feature Based Representations," *Winter Annual Meeting of ASME*, DE, vol. 21, San Francisco, ASME, 1989, pp. 21-34.

[Gair86] Gairola, A., "Design Analysis for Automatic Assembly," International Journal of Production Research, vol. 24, no. 4, 1986, pp. 839 - 849.

[Gu89] Gu, P.H., et al., "FDDL: A Feature Based Design Description Language," Design Theory and Methodology - DE, vol. 17, 1989, pp. 53 - 63.

[Holb88] Holbrook, A.E.K., and Sackett, P.J., "Design for Assembly - Guidelines for Product Design," Assembly Automation - *Proceedings 9th International Conference*, 1988, pp. 201 - 212.

[Hern91] Hernandez, J.A., et al., "An Integrated System for Concurrent Design Engineering," *Proceedings of the Seventh Conference on Artificial Intelligence Applications*, IEEE Computer Society, Feb. 1991, pp. 205-211.

[Hund90] Hundal, M.S., and Byrne, J.F., "Computer-Aided Generation of Function Block Diagrams in a Methodical Design Procedure," Design Theory and Methodology - DE, vol. 27, 1990, pp. 251 - 257.

[Ishi87] Ishii, K., "Active Constraint Deduction - A Framework For Expert Systems in Mechanical System Design," Advances in Design Automation, 1987, pp. 417 - 424.

[Jabl90] Jablokow, A.G. et al., "Topological and Geometric Consistency in Boundary Representations of Solid Models," Advances in Design Automation, vol. 23, no. 1, 1990, pp. 59 - 66.

[Jans90] Jansson, D.G., et al., "Generalized Measures of Manufacturability," Design Theory and Methodology - DE, vol. 27, 1990, pp. 85 - 96.

[Josh90] Joshi, S., and Chang T.C., "Feature Extraction and Feature Based Design Approaches in the Development of Design Interface for Process Planning," Journal of Intelligent Manufacturing, vol. 1, 1990, pp. 1 - 15.

[Just92] Juster N.P., "Modeling and Representation of Dimensions and Tolerances: A Survey," Computer Aided Design, vol. 24, no. 1, 1992, pp. 3 - 17.

[Kalp89] Kalpakjian, S., *Manufacturing Engineering and Technology*, New York, Addison-Wesley, 1989.

[Kari90] Karinthi, R., and Nau, D., "An Approach to Addressing Geometric Feature Interactions in Concurrent Design," Computers in Engineering, vol. 1, 1990, pp. 243 - 250.

[Karr90] Karra, C., and Phelps, T.A., "Tool Approach Directions for Machining Protrusion and Depression Features in an Object," Advances in Design Automation - DE, vol. 23 - 1, 1990, pp. 193 - 200.

[Keir90] Keirouz, W., Pabon, J., and Young, R., "Integrating Parametric Geometry, Features, and Variational Modeling for Conceptual Design," Design Theory and Methodology - DE, vol. 27, 1990, pp. 1 - 9.

[Kim89] Kim, J.J., and Gossard, D.C., "Reasoning on the Location of Components for Assembly Packaging," *Proceedings of the ASME Design Automation Conference*, Montreal, Canada, 1989, pp. 251-257.

[Kim91] Kim, H.S., Hom, S., and Parthasarathy, S., "Design and Manufacturing Advisor for Turbine Disks," Design For Manufacture - Strategic principles and techniques, (England: Addison-Wesley)

(1991), pp. 215 - 230.

[Kim92] Kim, Y.S., and Roe, K.D., "Conversions in Form Feature Recognition Using Convex Decomposition," Computers in Engineering, vol. 1, 1992, pp. 233 - 244.

[Kimu86] Kimura, F., and Suzuki, H., "Variational Product Design by Constraint Propagation and Satisfaction in Product Modeling," Annals of the CIRP, vol. 35, no. 1, 1986, pp. 75 - 78

[Kimu87] Kimura, F., et al., "Variational Geometry Based on Logical Constraints and its Applications to Product Modeling," Annals of the CIRP, vol. 36, no. 1, 1987, pp. 65 - 68

[Kond90] Kondo, K., "PIGMOD: Parametric and Interactive Geometric Modeler for Mechanical Design," Computer-Aided Design, vol. 22, no. 10, 1990, pp. 633 - 644.

[Kond92] Kondo, K., "Algebraic Method for Manipulation of Dimensional Relationships in Geometric Models," Computer-Aided Design, vol. 24, no. 3, 1992, pp. 141 - 147.

[Kota90] Kota, S., and Ward, A.C., "Functions, Structures, and Constraints in Conceptual Design," Design Theory and Methodology - DE, vol. 27, 1990, pp. 239 - 250.

[Lawl89] Lawlor-Wright, T., and Hannam, R.G., "A Feature-Based Design for Manufacture: CADCAM Package," Computer-Aided Engineering Journal, vol. 6, no. 6, 1989, pp. 215 - 220.

[Ligh82] Light, R., and Gossard, D., "Modification of Geometric Models Through Variational Geometry," Computer Aided Design, vol. 14, no. 4, 1982, pp. 209 - 214.

[Lin89] Lin, A.C., and Chang, T.C., "A Framework for Automated Mechanical Assembly Planning," Journal of Mechanical Working Technology, vol. 20, 1989, pp. 237 - 248.

[Liu84] Liu, C.R., and Srinivasan, R., "Generative Process Planning Using Syntactic Pattern Recognition," Computers in Mechanical Engineering, March 1984, pp. 63-66.

[McMa92] McMahon, et al., "Observations on the Application and Development of Parametric-Programming Techniques," Computer Aided Design, vol. 24, no. 10, 1992, pp. 541 - 546.

[Main92] Main, B.W., and Ward, A.C., "A Potential Framework for CAD/CAE/CIM Decisions," Computers in Engineering, vol. 1, 1992, pp. 61 - 67.

[Mant89] Mantalya, M., Opas, J., and Puhakka, J., "Generative Process Planning of Prismatic Parts by Feature Relaxation," Advances in Design Automation, vol. 19-1, 1989, pp. 417 - 424.

[Maye87] Mayer, R.J., "IGES - One Answer to the Problems of CAD Database Exchange," Byte, vol. 12 - 6, 1987, pp. 209 - 214.

[Meeh90] Meehan, E.J., and Brown, D.C., "Constraint Absorption and Relaxation Using a Design History," Design Theory and Methodology - DE, vol. 27, 1990, pp. 193 - 202.

[Miet91] Mietrach, D., "Advanced Castings in Aircraft Structures: A Way to Reduce Costs," Design For Manufacture - Strategic principles and techniques, (England: Addison-Wesley), 1991, 69-82.

[Nits91] Nitschke, D.R., Chen, Y.M., and Miller, R.A., "A Feature Extraction Interface for CAD Part Models," Computers in Engineering, vol. 1, 1991, pp. 367 - 373.

[Nnaj88] Nnaji, B.O., "A Framework for CAD-Based Geometric Reasoning for Robot Assembly Planning," The International Journal of Production Research, vol. 26, no. 5, 1988, pp. 735 - 768.

[NSF87] National Science Foundation, "Research Priorities for Proposed NSF Strategic Manufacturing Initiative," Report of a National

Science Foundation Workshop, National Science Foundation, Washington, DC, 1987.

[Ogra91a] O'Grady, P., and Young, R.E., "Issues in Concurrent Engineering Systems," Journal of Design and Manufacturing, vol. 1, 1991, pp. 27 - 34.

[Ogra91b] O'Grady, P., and Oh, J., "A Review of Approaches to Design for Assembly," Concurrent Engineering, vol. 1, 1991, pp. 5 - 11.

[Ogra91c] O'Grady, P., Young, R.E., Greef, A., and Smith, L., "An Advice System for Concurrent Engineering," International Journal of Computer Integrated Manufacturing, 4:63-70, 1991.

[Oh90] Oh, J.S., and O'Grady, P., "A Review of Approaches to Design for Assembly," LISDEM Technical Report TR-90-08, 1990.

[Oh91] Oh, J.S., O'Grady, P., and Young, R.E., "An Artificial Intelligence Constraint Network Approach to Design for Assembly," Department of Industrial Engineering Technical Report, 91-02, 1991.

[Orad89] Orady, E.A., "Design and Development of an Expert System for Evaluation of the Manufacturability of Mechanical Parts," *Proceedings of the 15th Annual AMSE Conference*, Brockport, New York, Association of Muslim Scientists and Engineers, Oct. 1989, pp. 51 - 66.

[Owen85] Owen, T., *Assembly With Robots*, Prentice-Hall Inc. Englewood Cliffs, New Jersey, 1985.

[Paol92] Paoluzzi, A., and Sansoni, C., "Programming Language for Solid Variational Geometry," Computer Aided Design, vol. 24, no. 7, 1992, pp. 349 -366.

[PDES88] PDES, Product Data Exchange Specification: First Working Draft, PB89-144794, Department of Commerce, Gaithersburg, MD, USA,

1988.

[PDES90] PDES/STEP, ISO TC184/SC4/WG1, 1990.

[Pham82] Pham, D.T., "On Design Components for Automatic Assembly," *Proceedings of the 3rd International Conference*, Stuttgart, Germany, 1982, pp. 205-214.

[Poli86] Poli, C., Graves, R., and Gropetti, R., "Rating Products for Ease of Assembly," Machine Design, August 1988, pp. 137 - 140.

[Pomr92] Pomrehn, L.P., and Papalambros, P.Y., "Constraint Activity Revisited: Application to Global and Discrete Design Optimization, Advances in Design Automation - DE , vol. 44 - 1, 1992, pp. 223 - 229.

[Pratt84] Pratt, M.J., "Solid Modeling and the Interface Between Design and Manufacture," IEEE Computer Graphics and Applications, vol. 4, no. 7, 1984, pp. 52 - 59.

[Rama91] Ramaswamy, R., et al., "Solving Parametric Design Problems Requiring Configuration Choices," Design Theory and Methodology, DE - vol. 31, ASME, 1991, pp.103 - 110.

[Rao91] Rao, M., and Luxhoj, J.T., "Integration Framework for Intelligent Manufacturing Processes," Journal of Intelligent Manufacturing, vol. 2, 1991, pp. 43 - 52.

[Requ82] Requicha, A.A.G., and Voelcker, H.B., "Solid Modeling: A Historical Summary and Contemporary Assessment," IEEE Computer Graphics and Applications, vol. 2, no. 2, 1982, pp. 9 - 24.

[Requ83] Requicha, A.A.G., and Voelcker, H.B., "Solid Modeling: Current Status and Research Directions," IEEE Computer Graphics and Applications, 1983, pp. 25 - 37.

[Requ89] Requicha, A.A.G., and Vandenbrande, J.H., "Form Features for

Mechanical Design and Manufacturing," Computers in Manufacturing, Vol. 1., 1989, pp. 47 - 52.

[Rind90a] Rinderle, J.R., and Krishnan, V., "Constraint Reasoning in Concurrent Design," Design Theory and Methodology - DE, vol. 27, 1990, pp. 53 - 62.

[Rind90b] Rinderle, J.R., and Balasubramaniam, L., "Automated Modeling to Support Design," Design Theory and Methodology - DE, vol. 27, 1990, pp. 281 - 290.

[Roll91] Roller, D., "An Approach to Computer-Aided Parametric Design," Computer Aided Design, vol. 22, no. 5, 1991, pp. 385 - 391.

[Rols88] Rolston, D.W., *Principles of Artificial Intelligence and Expert Systems Development*, New York, McGraw-Hill Book Company, 1988.

[Shah89] Shah, J.J., and Miller, D., "A Structure for Integrating Geometric Tolerances with Form Features and Geometric Models," Computers in Engineering, 1989, pp. 395 - 402.

[Shah91] Shah, J.J., and Mathew, A., “Experimental Investigation of the Step Form-Feature Information Model,” Computer Aided Design Journal, vol. 23, no. 4, 1991, pp. 282 - 296.

[Shah92a] Shah, J.J., and Tadepalli, R., "Feature Based Assembly Modeling," Computers in Engineering, vol. 1, 1992, pp. 253 - 260.

[Shah92b] Shah, J.J., et al., "Synergetic Design Systems," Computers in Engineering, vol. 1, 1992, pp. 283 - 290.

[Spil89] Spillers, W.R., and Newsome, S., "Design Theory: A Model for Conceptual Design," Design Theory '88, (New York: Springer-Verlag), 1989.

[Srih92] Srihari, K., and Emerson, C.R., "A Concurrent Engineering Methodology for PCB Assembly," Khalil TM, Bayraktar BA (eds.):

Management of Technology III, Institute of Industrial Engineers, 1992, pp. 828 - 837.

[Srik92] Srikantappa, A.B., and Crawford, R.H., "Intermediate Geometric and Interfeature Relationships for Automatic Group Technology Part Coding," Computers in Engineering, vol. 1, 1992, pp. 245 - 251.

[Srin85] Srinivasan, R., and Liu, C., "Extraction of Manufacturing Details from Geometric Models," Computers and Industrial Engineering, Vol. 9, No. 2, 1985, pp. 125-133.

[Srin89] Srinivasan, S., and Allen, R.H., "Partitioning and Guided Search: A generalized Approach to Problem Solving in Preliminary Design," Design Theory and Methodology - DE, vol. 17, 1989, pp. 37 - 43.

[Stau89] Stauffer, L.A., and Slaughterbeck-Hyde, R.A., "The Nature of Constraints and Their Effect on Quality and Satisficing," Design Theory and Methodology - DE, vol. 17, 1989, pp. 1 - 7.

[Subr89] Subramanyam, S., and Lu, S.C., "Computer-Aided Simultaneous Engineering for Components Manufactured in Small and Medium Lot Sizes," DE, vol. 21, PDE vol. 36, ASME, 1989, pp. 175 - 184.

[Taka86] Takahashi, K., and Senba, K., "Design for Automatic Assembly," Assembly Automation, 7th International Conference, 1986, pp.

[Truc87] Trucks, H.E., *Designing for Economical Production*, Published by SME, Dearborn, Michigan, 1987.P

[Tsan89] Tsang, J.P., and Brissaud, D., "A Feature-Based Approach to Process Planning," Computers in Engineering, 1989, pp. 419 - 430.

[Ulri90] Ulrich, K.T., and Graham, P.V., "Using Producibility Constraints to Control the Automatic Generation of Sheet Metal Structures," Design Theory and Methodology - DE, vol. 27, 1990, pp. 97 - 104.

[Vand90] Vandenbrande, J.H., and Requicha, A.A.G., "Spatial Reasoning for Automatic Recognition of Interacting Form Features," Computers in Engineering, vol. 1, 1990, pp. 251 - 256.

[Vass92] Vasseur, H., Kurfess, T.R., and Cagan, J., "A Decision-Analytic Method for Competitive Design for Quality," Advances in Design Automation - DE , vol. 44 - 1, 1992, pp. 329 - 336.

[Verr92] Verroust, A., Schonek, F., and Roller, D., "Rule-Oriented Method for Parametrized Computer-Aided Design," Computer-Aided Design, vol. 24, no. 10, 1992, pp. 531 - 540.

[Wald89] Waldron, M.B., Waldron, K.J., and Owen, D.H., "Use of Systemic Theory to Represent the Conceptual Mechanical Design Process," Design Theory '88, (New York: Springer-Verlag), 1989.

[Wang87] Wang, H., and Wysk, R.A., "Intelligent Reasoning for Process Planning," Computers in Industry, No. 8, 1987, pp. 293-309.

[Wang92] Wang, E., "Using Automatic Feature Recognition to Interface CAD to CAPP," Computers in Engineering, vol. 1, 1992, pp. 215 -231.

[Watt90a] Watton, J.D., and Rinderle, J., "Improving Mechanical Design Decisions with Alternate Formulations of Constraints," Design Theory and Methodology - DE, vol. 27, 1990, pp. 69 - 75.

[Watt90b] Watton, J.D., and Rinderle, J., "A Method to Identify Reformulations of Mechanical Parametric Constraints to Enhance Design," Design Theory and Methodology - DE, vol. 27, 1990, pp. 77 - 84.

[West91] Weston, R.H., et al., "'Soft' Integration and its Importance in Design to Manufacture," Journal of Design and Manufacturing, vol. 1, 1991, pp. 47 - 56.

[Wils85] Wilson, P.R., "Euler Formulas and Gemetric Modeling," IEEE - Computer Graphics and Applications, vol. 5, No.8, 1985, pp. 24 -

36.

[Wils88] Wilson, P.R., and Pratt, M.J., "A Taxonomy of Features for Solid Modeling," Geometric Modeling for CAD Applications, (Wozny, M.J., et al - editors), Elsevier Science Publishers B.V., 1988, pp. 125 - 136.

[Wohl92] Wohlsen, R.C., "Interaction-Based Schematics for Representing Mechanical Systems," Computers in Engineering, vol. 1, 1992, pp. 83 - 89.

VITA

Krishna Kumar Krishnan was born on June 6, 1961 in Trivandrum, India. On completing his Bachelors in Mechanical engineering from Kerala University in August 1984, he joined the Indian Space Research Organization as a Scientific Officer. He was primarily involved with the research and development of new manufacturing technologies and the creation of quality control procedures for the first liquid engine booster ever to be developed by India. To pursue higher studies he resigned from his job and began his graduate work in Industrial and Systems Engineering at Virginia Tech in the Fall of 1988. He completed his M.S. in 1991. Upon the completion of his Ph.D. he will be joining the Industrial Engineering department at Cleveland State University, Cleveland, Ohio.