Fall 1991

Computer-aided process planning (CAPP) SORICH

Sohail Pirzada
New Jersey Institute of Technology

ABSTRACT

Title: COMPUTER-AIDED PROCESS PLANNING (CAPP) - SORICH

Author: SOHAIL PIRZADA

Degree: Master of Science

Date: December, 1991

Advisor: Steve Kotefski

Department: Manufacturing Engineering

The last two decades have witnessed the increasing role of the computer in the process planning function. This has been further enhanced by the advent of the knowledge based expert system, and its impact on Computer Aided Process Planning (CAPP). CAPP has emerged as a strategic link between design and manufacture. This thesis discusses the various methods used in the process planning function. Group Technology (GT) plays a pivotal role in establishing CAPP. The utilization of Artificial Intelligence techniques in CAPP is listed. This study also presents an Interactive Software adaptive to product/part configuration variables.

COMPUTER-AIDED PROCESS PLANNING
(CAPP)
SORICH

by

SOHAIL PIRZADA

Thesis Submitted To The Faculty Of Graduate Division Of The
New Jersey Institute of Technology
In Partial Fulfillment Of The Requirement For The Degree Of
Master Of Science
Manufacturing Engineering

December, 1991

APPROVAL SHEET

Title of The Thesis: **COMPUTER-AIDED PROCESS PLANNING (CAPP) - SORICH**

Name of The Candidate: **SOHAIL PIRZADA**

Thesis and Abstract Approved | **Date Approved**

Steve Kotefski
Professor
Department of Manufacturing
Engineering Technology

Dr. Raj Sodhi
Professor
Department of Manufacturing
Engineering

Nouri Levy
Professor
Department of Mechanical
Engineering

VITA SHEET

Name: **SOHAIL PIRZADA**

Degree & Date: Master of Science, December, 1991

Education:

Institute	Degree	Major	Date
NED University of Engineering & Technology, Karachi, Sindh - Pakistan	B.E (B.S)	Mechanical	1989
New Jersey Institute of Technology, New Jersey	M. S	Manufacturing	1991

To

my

MOTHER

Nasim Akhtar Pirzada

ACKNOWLEDGEMENT

The author wishes to express his sincere gratitude to Professor Steve Kotefski of Manufacturing Engineering Technology Department at New Jersey Institute Of Technology for his assistance, guidance and encouragement as Graduate Advisor.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

1. Computer Aided Process Planning (CAPP)

A common task practiced in most manufacturing industries is process planning. Process planning is the function within a manufacturing facility that establishes which processes and parameters are to be used (as well as those machines capable of performing these processes) to convert a piece part from its initial form to a final form predetermined (usually by a design engineer) in an engineering drawing. Alternatively, process planning could be defined as the act of preparing detailed work instructions to produce (machine or assemble) a part. Required inputs to the process planning are:

* Geometric Features
* Dimensions
* Tolerances
* Materials
* Finishes.

These inputs are analyzed and evaluated in order to select an appropriate sequence of processing operations based upon specific, available machinery and workstations.

A process plan needs to generate operational details or outputs such as:

* Operation sequence
* Feeds
* Depth of cut
* Required tooling
* Machine tools
* Standard times and costs
* Workpiece holding devices(jigs)
* Material Removal Rate

In general, a process plan is prepared using available design data and manufacturing knowledge. The process planning function has been described as "a subsystem that is responsible for the conversion of design data to work instructions".

A major thrust has been to promote the use of Computer-Aided Process Planning (CAPP) as a necessary, logical interface between Computer-Aided Design (CAD), and Computer-Aided Manufacturing (CAM).

Current CAD systems tend to concentrate on automated design, solids modeling, drafting, and tool path generation for numerically controlled machine tools.

CAM systems tend to concentrate on automated manufacturing and robot usage. Integration of these two procedures could be brought about ideally through the use of CAPP.

Softwares are available to perform these tasks, but at quite an expense. To demonstrate the theory of CAPP, the software package, Quattro-Pro was customized to produce a route sheet and bill of materials. The products of the Intrex Corporation (a custom furniture company) were used as a model for this thesis. Most CAPP systems are component oriented. Keeping the Intrex Corporation in mind (which is a product oriented company) we designed our CAPP system to function on a product oriented basis. However, the system can be adapted to serve as a component based CAPP system.

CHAPTER 2

2. Process Planning Methods

There are essentially three possible approaches to process planning:

* Manual
* Computer-assisted variant
* Computerized Generative

It is stated by H.J. Steudel that it may be more appropriate to use a particular approach under a given set of circumstances.

Closely related to process planning are the functions of determining appropriate cutting conditions for the machining operations and setting the time standards for the operations. All three functions:

* Planning the process
* Determining the cutting conditions and
* Setting the time standards

These tasks have traditionally been carried out as tasks with a very high manual content. They are also typically routine tasks in which similar or even identical decisions are repeated over and over. Today, these kinds of decisions are being made with the aid of computers.

The initial material can take a number of forms, the most common of which are bar stock, plate, castings, forgings, or maybe just a slab (of any geometry). The slab of material is normally a burnout cut to some to some rough dimensions. This metal slab can consist of almost any geometry. The selection of raw stock requires some study to ensure its specified engineering properties and economical viability.

With these raw materials as a base, the process planner must prepare a list of processes to convert this normally predetermined material into a predetermined final shape. The processes used by a process planner in a discrete-part metal-manufacturing industry are listed in Table-1. Some of the operations given in Figure 1 are often considered subsets of some major category (e.g., facing can be considered a subset of turning and reaming can be considered a subset of drilling).

The process plan in Figure 2 (Intrex Process plan) is sometimes called an operation sheet, route sheet, or operation planning summary. A detailed plan usually contains the route, processes, process parameters , and machine and tool selections. In a more general sense, a process is

TABLE 1. OPERATIONS AND MACHINES FOR THE MACHINING OF SURFACES

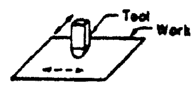
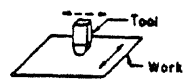
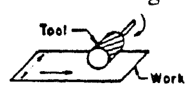
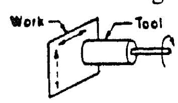
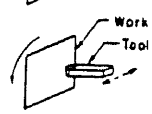
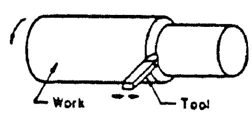
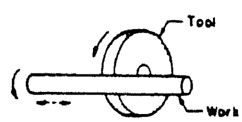
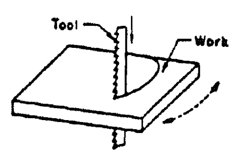
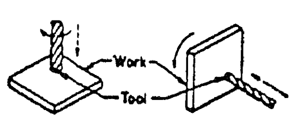
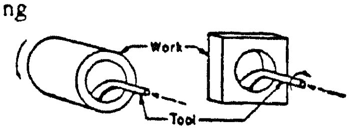
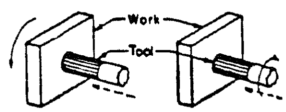
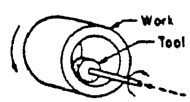
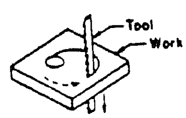
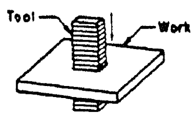
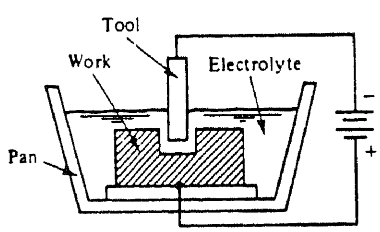
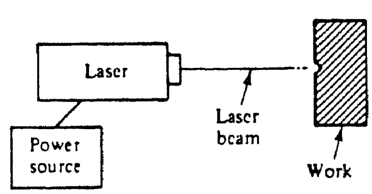
Operation	Block Diagram	Most Commonly Used Machines	Machines Less Frequently Used	Machines Seldom Used
Shaping		Horizontal shaper	Vertical shaper	
Planing		Planer		
Milling	slab milling  face milling 	Milling machine		Lathe (with special attach-ment)
Facing		Lathe	Boring mill	
Turning		Lathe	Boring mill	Vertical shaper Milling machine
Grinding		Cylindrical grinder		Lathe (with special attach-ment)
Sawing		Contour saw		
Drilling		Drill press	Lathe	Milling machine Boring mill Horizontal boring machine

TABLE 1 (Continued).

Operation	Block Diagram	Most Commonly Used Machines	Machines Less Frequently Used	Machines Seldom Used
Boring		Lathe Boring mill Horizontal boring machine		Milling machine Drill press
Reaming		Lathe Drill press Boring mill Horizontal boring machine	Milling machine	
Grinding		Cylindrical grinder		Lathe (with special attach-ment)
Sawing		Contour saw		
Broaching		Broaching machine		
ECM			ECM machine	
Laser			CO_2 laser YAG laser	

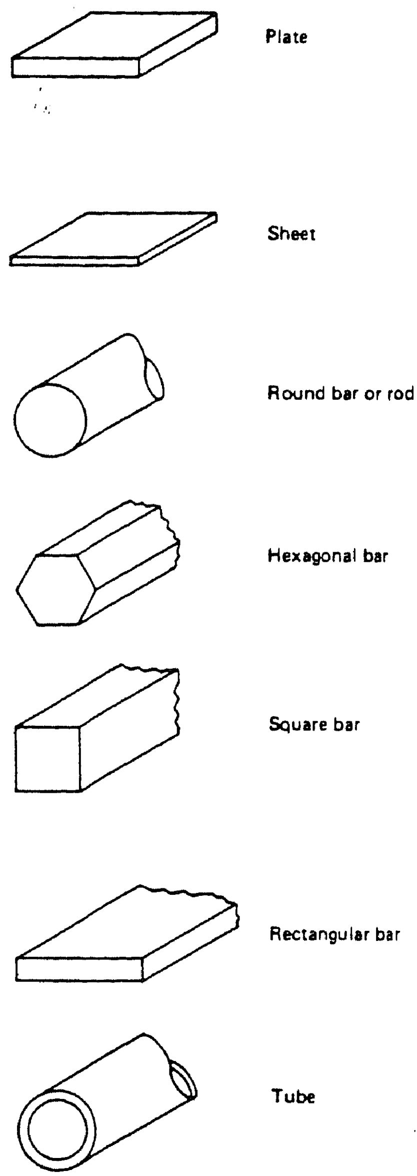


FIGURE 1. Standard Material Shapes.

called an operation (including manual operations). The route is the operation sequence. The process plan provides the instructions for the production of the part. These instructions dictate the cost, quality, and rate of production ; therefore, process planning is of utmost importance to the production system.

In a conventional production system, a process plan is created by a process planner, who examines a new part (engineering drawing) and then determines the appropriate procedure to produce it. The previous experience of the process planner is critical to the success of the plan. Planning, as practiced today, is as much an art as it is a formal procedure.

As mentioned previously, there are numerous factors that effect process planning. The design (shape), dimensions, tolerance, surface finish, size, material type, quantity, and the manufacturing system itself all contribute to the selection of operations and the operation sequence.

In addition to operation sequencing and operation selection, the selection of tooling, jigs and fixtures (sometimes the design of jigs and fixtures) is also a major part of the process planning function. The tooling portion includes selection of both the tool itself and the machine on which the tool is used. There is a limited set of commercially available tools, with different shapes, diameters, lengths,

Intrex Corporation

Manufacturing Methods(Process Plan)

Family: The Taper Series

Description: 54" Conference Table

Part: Apron

Department: Table

Note: All dimensions are final dimensions after the operation.

#	Operation Description	Machine/ Equipment	Setup Description
1	Select 4 strips for 4 aprons	Square	Check if strips are twisted standing on edges, if twisted more than 1/32" over 3" in width, select other strip(s).
2	Trim 4 strips to Width: - 3"	Table Saw	Use 10" dia., 60 teeth,TCG(triple chip grind) Carbide tipped circular saw blade. **Note:** Trim both edges of strip.
3	Cut 4 strips to Length: - 26 13/16"	Table Saw	42" dia. insert, 42" stopper, jig # j-aa196. Lay strip on insert. Use 10" dia., 60 teeth, TCG Carbide tipped circular saw blade. Slide strip(s) on track of insert to check if strip is twisted, if twisted select other strip(s). **Note:** Trim both ends.

FIGURE 2. Process Plan.

numbers of teeth, and alternative tool materials. The most commonly used tool materials are high speed steel (HSS) and tungsten carbide (WC). In recent years, ceramic tools have also become reasonably popular. WC can machine metal at a higher cutting speed than HSS, but costs more and is difficult or impossible to regrind. WC is also brittle and is not recommended for interrupted cuts. All these conditions have to be taken into account to select an appropriate tool.

Jigs and fixtures are devices to guide a tool or hold a workpiece for better machining. Very few standard devices are available. Even with standard devices, application is normally left to the machine operator. Process planning, however, should consider the impact of tooling and jigs and fixtures on the quality of the product during the selection of the production operations.

Some of the prerequisites for the process planners are given.

2.1. Engineering Design

Engineering design is the partial realization of a designer's concept. The designer normally cannot directly transform a concept into a physical item. Instead, the designer conveys the idea to others through an alternative medium, such as an engineering drawing, and then the manufacturing engineer or machinist produces the design. When a farmer needed a tool prior to the industrialized age, he normally went to a blacksmith and told the blacksmith the shape and size of the tool required. Because most tools were simple and did not require significant accuracy , the blacksmith would get a pretty good picture of a hoe or a plow through the verbal description. If the blacksmith still did not understand, the farmer could sketch the tool on the dirt floor of the blacksmith shop.

As product design requirements became more complex, a picture became necessary to relate the information to others. Multiview orthographic drawing have long been adopted by engineers as the standards tool to represent a design. Design information can be passed from the designer to others who are well trained in reading such drawings. The object a designer draws on paper can be interpreted and reconstructed in a viewer's mind. The capability of transforming an object from one medium to another (e.g., from a two-dimensional three view drawing to a three-dimensional picture) and an understanding of the rules of drawing are prerequisite to pass design information from the designer to others without error.

There are several methods available to represent an engineering drawing. The conventional method is drafting on

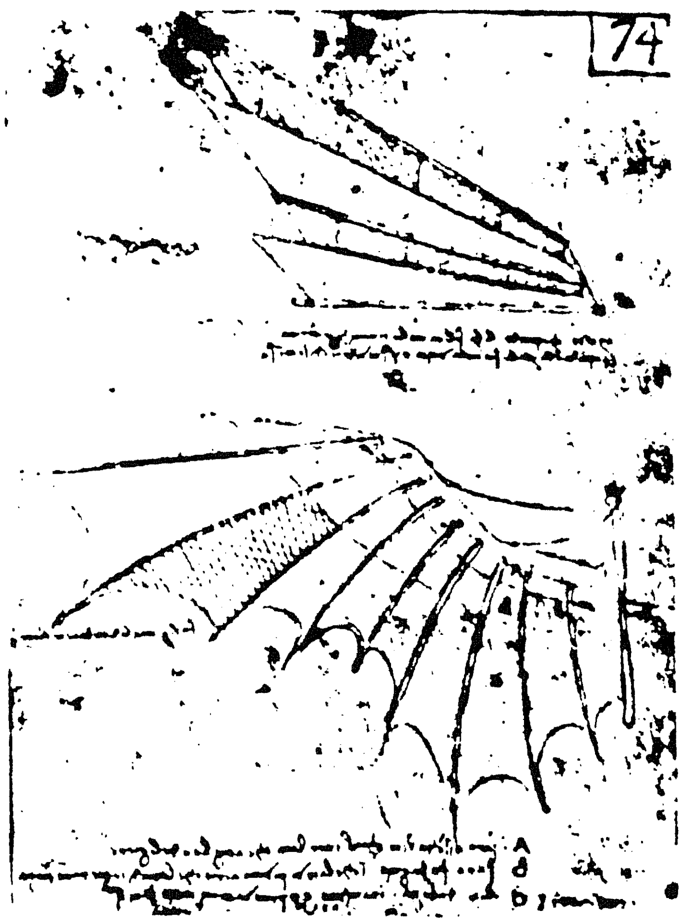


FIGURE 3. Idea Sketch Prepared By Leonardo da Vinci (1452-1519).

paper with pen or pencil. Manual drafting is tedious and requires a tremendous amount of patience and time. Recently, computer-aided drafting systems have been implemented to improve drafting efficiency. The major objective of these systems is to assist the draftsman with tedious drawing and redrawing. Partially completed or completed drawings from a graphics tablet or screen can be stored in a computer and retrieved when needed.

Engineering drawing is a universal language used to represent a designer's idea to others. It is the most accepted medium of communication in all phases of industrial and engineering work. In ancient times, before multiview drawing standards/concepts were adopted, perspective drawings were normally employed. The great master of art during the Renaissance, Leonardo da Vinci, designed several machines and mechanical components (which still amazes the contemporary designers) using perspective sketches, Figure 3. Today, pictorial drawings are still used to supplement other design representation.

2.2. Dimensioning

A design is expected to convey a complete description of every detail of a part. However, dimensioning is as important as the geometric information. It saves money directly by providing for maximum producibility of the part through maximum production. It provides uniformity and convenience in drawing delineation and interpretation, thereby reducing controversy and guesswork.

According to the American National Standards Institute (ANSI) standards, the following are the basic rules that should be observed in dimensioning any drawing:

2.2.1. Show enough dimensions so that the intended sizes and shapes can be determined without calculating or assuming any distances.

2.2.2. State each dimension clearly, so that it can be interpreted in only one way.

2.2.3. Show the dimensions between points, lines, or surfaces that have a necessary and specific relation to each other or that control the location of other components or mating parts.

2.2.4. Select and arrange dimensions to avoid accumulations of tolerances that may permit various interpretations and cause unsatisfactory mating of parts and failure in use.

2.2.5. Show each dimension only once.

2.2.6. Where possible, dimension each feature in the view where it appears in profile, and where its true shape appears.

2.2.7. Wherever possible, specify dimensions to make use of readily available materials, parts, tools, and gages. Savings are often possible when drawings specify:

a. Commonly used materials in stock sizes

b. Parts generally recognized as commercially standard

c. Sizes that can be produced with standard tools and inspected with standard gages

d. Tolerances from accepted published standards.

2.3. Tolerances

Since it is impossible to produce the exact dimension specified, a tolerance is used to show the acceptable variation in a dimension. The higher the quality a product has, the smaller the tolerance value specified. Tighter tolerances are translated into more careful production procedures and more rigorous inspection. It ensures that design dimensional and tolerance requirements, as they relate to actual function, are specifically stated and thus carried out. It ensures interchangeability of mating parts

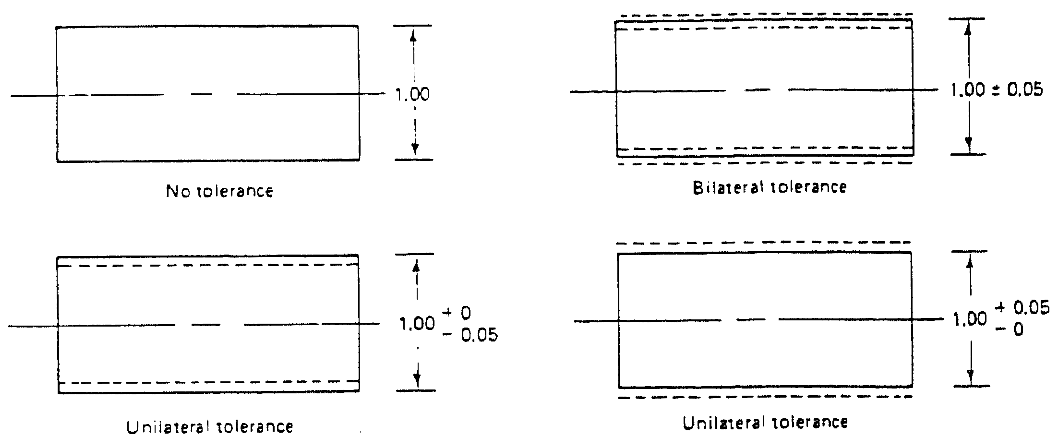


FIGURE 4. Tolerancing: Bilateral And Unilateral. Dashed Lines Show The Tolerance Limits.

at assembly. It is increasingly becoming the "spoken word" throughout industry, the military, and, internationally, on engineering drawing documentation. Every engineer or technician involved in originating or reading a drawing should have a working knowledge of this new state of the art.

There are two types of tolerances:

2.3.1. Bilateral tolerance

2.3.2. Unilateral tolerance

Figure 4 shows bilateral and unilateral tolerance. Unilateral tolerances, such as $1.00^{+0.00}_{-0.05}$, specify dimensional variation from the basic size (i.e., decrease) in one direction in relation to the basic size; for example,

$$1.00^{+0.00}_{-0.05} = 0.95 \text{ to } 1.00$$

The basic location where most dimension lines originate is the reference location, cumulative errors can be eliminated. Most mechanical parts contain both working surfaces and nonworking surfaces. Working surfaces are those for items such as bearings, pistons, and gear teeth, for which optimum performance may require control of the surface characteristics. Nonworking surfaces, such as the exterior walls of an engine block, crankcase, or differential housings, seldom require surface control. For surfaces that require surface control, control surface symbols can be used. Figure 5 shows how these symbols are used.

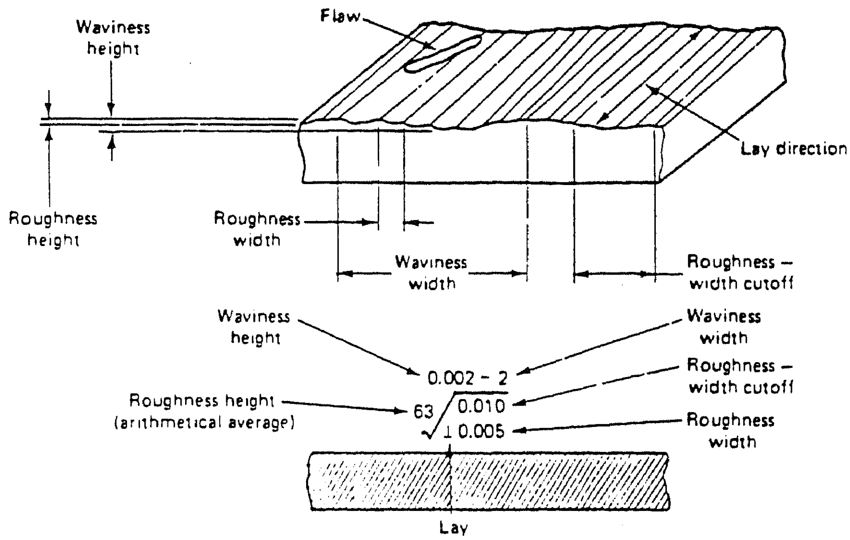


FIGURE 5. Surface Control Symbols.

In the symbol, several surface characteristics are specified. The roughness height is the roughness value as normally related to the surface finish. It is the average amount of irregularity above or below an assumed center line. It is expressed in microinches (micro inch = 0.000001 in.) or, in the metric system, in micrometer (micro meter = 0.000001 m). Recommended roughness heights are given in Table 2. Lay is another property of machined surface. It is the direction of the predominant surface pattern, produced by tool marks or grain of the surface, ordinarily dependent upon the production methods used. Lay symbols are listed in Figure 6 .

TABLE 2. RECOMMENDED HEIGHT VALUE

Roughness value (μin.)	Type of surface	Purpose
1000	Extremely rough	Used for clearance surfaces only where good appearance is not required
500	Rough	Used where vibration, fatigue, and stress concentration are not critical and close tolerances are not required
250	Medium	Most popular for general use where stress requirements and appearance are essential
125	Average smooth	Suitable for mating surfaces of parts held together by bolts and rivets with no motion between them
63	Better-than-average finish	For close fits or stressed parts except rotating shafts, axles, and parts subject to extreme vibration
32	Fine finish	Used where stress concentration is high and for such applications as bearings
16	Very fine finish	Used where smoothness is of primary importance, such as high-speed shaft bearings, heavily loaded bearings, and extreme tension members
8	Extremely fine finish produced by cylindrical grinding, honing, lapping, or butting	Use for such parts as surfaces of cylinder
4	Superfine finish produced by honing, lapping, buffing, or polishing	Used on areas where packings and rings must slide across the surface where lubrication is not dependable

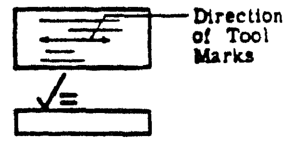
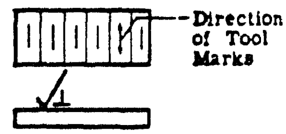
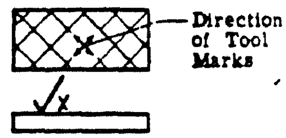
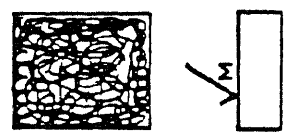
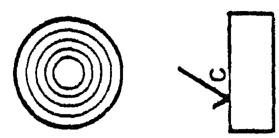
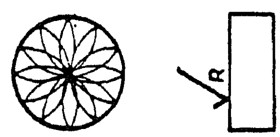
LAY SYMBOL	DESIGNATION	EXAMPLE
=	Lay parallel to the boundary line representing the surface to which the symbol applies.	
⊥	Lay perpendicular to the boundary line representing the surface to which the symbol applies.	
X	Lay angular in both directions to boundary line representing the surface to which symbol applies.	
M	Lay multidirectional.	
C	Lay approximately circular relative to the center of the surface to which the symbol applies.	
R	Lay approximately radial relative to the center of the surface to which the symbol applies.	

FIGURE 6. Lay Symbols.

2.4. Geometric Tolerances

Conventional methods of dimensioning only provide information concerning size and surface condition. A component can be produced without a guarantee of interchangeability. For example, in Figure 7, both components (b) and (c) satisfy the dimension specified in (a), that is, the diameter of components (b) and (c) is 0.501 in. over the entire length of the component. Obviously, both (b) and (c) are not desirable. However, as specified, both (b) and (c) meet specifications.

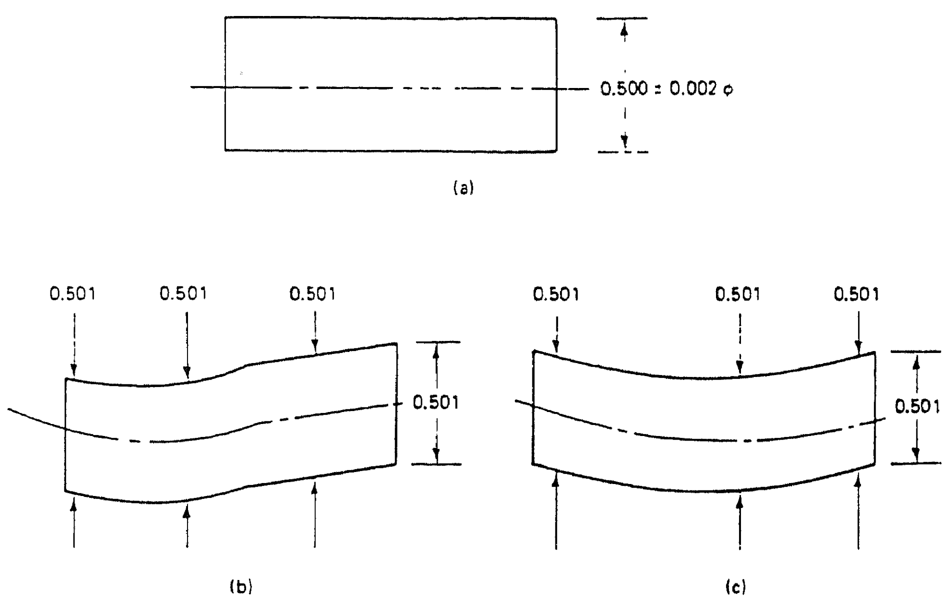
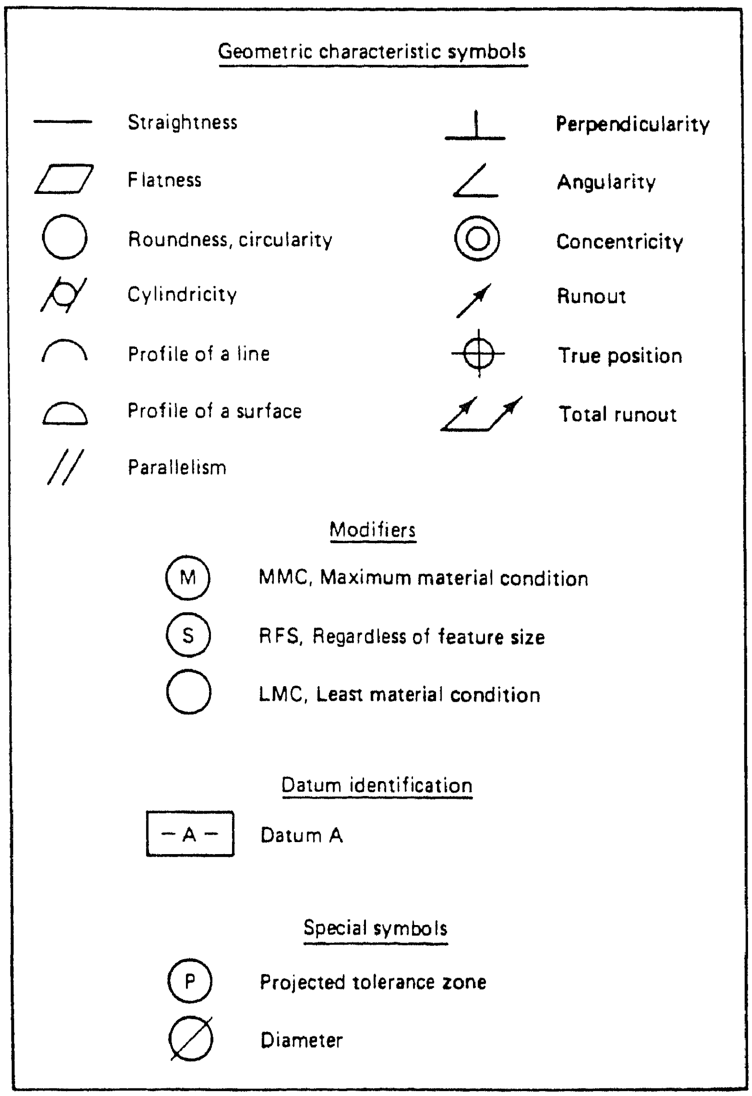
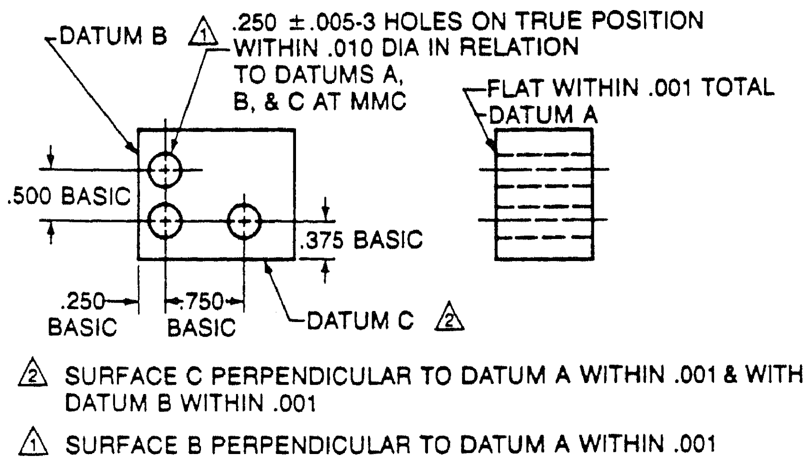


FIGURE 7. Illustration Of Some Addition Part Conditions.

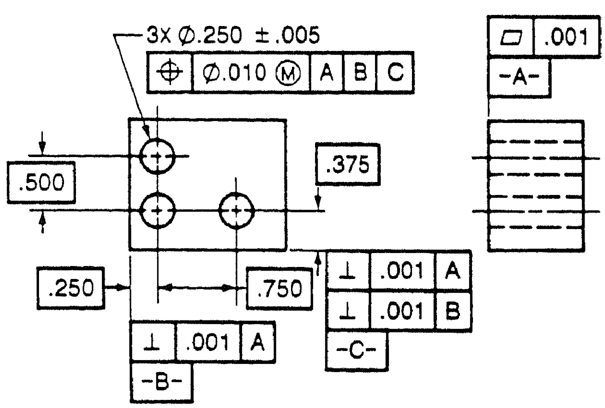
Geometric Tolerances specifies the tolerance of geometric characteristics. Basic geometric characteristics as defined by the ANSI Y14.5M 1982 standard that are used as the building blocks for geometric dimensioning and Tolerances are given in Table 3.

TABLE 3. GEOMETRIC TOLERANCING SYMBOLS





Using Notes



Using Symbols

FIGURE 8.

The general use of symbols instead of notes on a drawing provides a number of advantages. Figure 8 below incorporate the geometric characteristic symbols with datum and feature control symbols. Some of the advantages of symbols over notes are:

2.4.1. The symbols have uniform meaning. A note can be stated inconsistently, with a possibility of misunderstanding.

2.4.2. Symbols are compact, quickly drawn, and can be placed on the drawing where the control applies; symbols adapt readily to computer applications.

Notes require much more time and space, tend to be scattered on the drawing, often appear as footnotes which separate the note from the feature to which it applies.

2.4.3. Symbols are the international language and surmount individual language barriers.

Notes may require translation if the drawing is used in another country.

2.4.4. Symbols can be applied with drafting templates or computer techniques and retain better legibility in various forms of copy reproduction.

2.4.5. Geometric Tolerances symbols follow the established precedent of other well-known symbol systems, e.g., electrical and electronic, welding, surface texture, etc.

To specify the geometric tolerances, reference features--either planes, lines, or surfaces--can be established:

2.5. Maximum Material Condition (MMC)

One of the fundamental and most important of geometric dimensioning and Tolerances is Maximum Material Condition. The condition in which a feature of size contains the maximum amount of material within the stated limits of size: for example, minimum hole diameter and maximum shaft diameter.

2.6. Regardless of Feature Size (RFS)

The term used to indicate that a geometric tolerance or datum reference applies at any increment of size of the feature within its size tolerance. Unlike maximum material condition, the "regardless of feature size" principle permits no additional positional, form or orientation tolerance, no matter to which size the related features are produced. It is really the independent form of dimensioning and Tolerances which has always been used prior to the introduction of the MMC principle.

2.7. Least Material Condition (LMC)

The term used is the condition in which a feature of size contains the least amount of material within the stated limits of size, for example, maximum hole diameter, minimum shaft diameter. LMC is the condition opposite to MMC. For example, a shaft is at least material condition when it is at its high limit of size.

2.8. Datum

A theoretically exact point, axis, or plane derived from the true geometric counterpart of a specified datum feature. A datum is the origin from which the location or geometric characteristics of features of a part are established. Datum surfaces and datum features are actual part surfaces or features used to establish datums. They include all the surface or feature inaccuracies.

2.9. Basic

A numerical value used to describe the theoretically exact size, profile, orientation, or location of a feature or datum target. It is used as the basis from which permissible variations are established by tolerances on other dimensions or in notes.

Use of a Basic dimension, which is a theoretical and exact value, requires also tolerance stating the permissible variation from this exact value (most often relative to a position angularity or profile requirement). A basic dimension states only half the requirement. To complete it, a tolerance must be associated with the features involved in the Basic dimension.

2.10. Machine Tooling

The Machine Tooling portion includes selection of both the tool itself and the machine on which the tool is used. Machine tooling refers to the selection and design of cutting tools, jigs, fixtures, and dies required to perform a specific operation. This field, which is generally called tool engineering, has much to do with the operating efficiency of machines. A decision as to whether an air chuck or a mechanical chuck should be used; whether a cemented carbide tool or one made from high-speed steel will perform better; what speeds and feeds produce the most satisfactory results; whether water, oil, or air will serve best as a coolant - these are typical problems encountered by the process planner.

2.11. Jigs

Jigs are fixtures used in production drilling, tapping, boring, and reaming operations. Their function is:

2.11.1. To reduce the cost of operation.

2.11.2. To increase production

2.11.3. To assure high accuracy

2.11.4. To provide interchangeability

A jig must hold the part and guide the cutting tool. Such a device is a good example of the transfer of skill from a mechanic to an accessory part of a machine; thus the operation may be performed accurately by an unskilled operator. This is well illustrated in the drilling of four holes in a plate by using a plate or channel jig similar to the one shown in Figure 9. The jig is made with hardened steel bushings which locate the positions of the four holes accurately. Any number of plates may be clamped to this jig, and each part will be identical with the other.

Jigs perform the same function but differ widely in appearance according to the shape and design of the part to

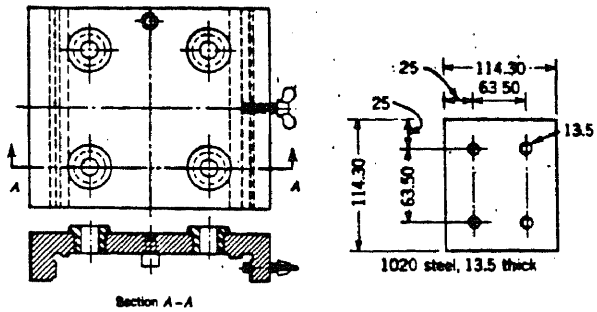


FIGURE 9. Plate Drilling Jig.

be worked on. Classification is based on their general appearance and construction.

A jig should be designed to provide quick and easy loading and unloading. Likewise, clamping devices must be positive, and the design should be such that there is no question about the proper location of the part in the jig. Clearance is usually provided under drill bushings to allow chips to escape without having to go through them. This is important if much metal is to be removed. In addition, provision should be made for rapid cleaning of chips from the jig. Most jigs utilize standard parts such as drill bushings, thumb screws, jig bodies, and numerous other parts. Jigs are not limited to drilling operations but are also used on tapping, counterboring, and reaming operations.

2.12. Fixtures

A fixture is a work-holding and work-supporting device that is securely held or fixed to a machine. Fixture, unlike jigs, do not guide the tool. Their primary function is:

2.12.1. To reduce the cost of operation

2.12.2. To increase production

2.12.3. To enable complex-shaped, and often heavy, parts to be machined by being held rigidly to a machine.

Fixtures are used most on lathes, turret lathes, milling machines, boring equipment, shapers, and planers.

Fixtures to hold a particular part are made from a grey cast iron or from steel plate by welding or bolting. It is fixed-that is bolted, clamped, or "set" with a low-melting alloy-to the machine. A fixture has the locating pins or machined blocks against which the workpiece is tightly held by clamping or bolting. In order to assure interchangeability, the locating devices are made from hardened steel. Many fixtures are massive because they, like a machine frame, may have to withstand large dynamic forces. Since all fixtures are between the workpiece and the machine, their rigidity and the rigidity of their attachment to the machine are paramount. Duplex fixtures have been built to allow for the loading or unloading of one side of fixture while the machining operation is taking place on a part clamped to the other side.

2.13. Dies

Dies for both the hot and cold-chamber machines are similar in construction because there is little difference in the method of holding and operating them. They are made in two sections to provide a means of removing the castings and are usually equipped with heavy dowel pins to keep the halves in proper alignment. Metal enters the stationary side when the die is locked in closed position. As the die opens, the ejector through the die force the casting from the cavity and fixed cores. The dies are provided with a separate mechanism for moving the ejector plate or movable cores. The life of these molds depends on the metal cast and may range from 10,000 fillings, if brass castings are made, to several million if zinc is used.

It is always desirable to provide vents and small overflow wells on one side of a die to facilitate the escape of air and to catch surplus metal that has passed through the die cavity. In spite of this provision, there is a certain amount of flash metal that must be trimmed off in the finishing operation.

For large or complex castings, a single-cavity mold is used. If the quantity of castings to be produced is large and they are relatively small in size, a multiple-cavity die can be used. A combination die is one that has two or more cavities, each of which is different. They are frequently made up of insert blocks that can be removed so that die blocks can be substituted. Most dies are provided with channels for water cooling to keep the die at the correct temperature for rapid production.

CHAPTER 3

3. Economical Production Criteria

The increasing need for finished goods in large quantities has, in the past, led engineers to search for and develop new methods for production. Many individual developments in the various branches of manufacturing technology have been made and have allowed the increased production of improved finished goods at lower cost. The cost of a product is based on expenditures for raw materials, machines, labor, sales, warehousing, and overhead. Machine and labor costs are inexorably related and make up, along with raw materials expenditures, the bulk of production costs. When a material is chosen, the process, including the machine, may be automatically specified, or if a machine is available, the raw material that can be used may be limited. One could say that the purpose of economical production is to produce a product at a profit. This infers that the cost must be acceptable and competitive, also, a demand for the product must exist, or else it must be created.

Since the first use of machine tools, there has been a gradual, but steady, trend toward making machines more efficient by combining time and labor. To meet these needs, machine tools have become complex both in design and in control. Automatic features have been built into many machines, and some are completely automatic. This technical development has made it possible for industry to attain a high production rate with the accompanying low labor cost that is an essential development for any society wishing to enjoy high living standards.

3.1. Automation

Automation was first coined by the automotive industry to describe methods used for automatically controlling various machines and the conveying of workpieces between these machines. Automation is a technology concerned with the application of mechanical, electronic, and computer-based systems to operate and control production. Coordination is not necessarily automation. Automation implies self-correction sch as the governor controlling the speed of an engine or the thermostat controlling the temperature in a house. On a machine doing a turning operation the feedback control would feed the tool in slightly if the diameter became too large. In other words, automation gives to the machine some power of choice or ability to correct itself if certain prescribed limits are exceeded. Automation is very tricky. One can't be successful just by automating the facility. This technology includes:

* Automatic machine tools to process parts

* Automatic assembly machines

* Industrial robots

* Automatic material handling and storage systems

* Automatic inspection systems for quality control

* Feedback control and computer process control

* Computer systems for planning, scheduling

* Computer systems for data collection and decision making to support manufacturing activities.

3.1.1. Types of automation

Automation production systems in manufacturing can best be classified into three basic types:

3.1.1.1. Fixed Automation

3.1.1.2. Programmable automation

3.1.1.3. Flexible automation

3.1.1.1. Fixed Automation is a system in which the sequence of processing (or assembly) operations is fixed by the equipment configuration. The typical features of fixed automation are:

* High initial investment for custom-engineered equipment

* High production rates

* Relatively inflexible in accommodating product changes

The economic justification for fixed automation is found with very high demand rates and volumes. Examples of fixed automation include mechanized assembly lines (starting around 1913-- the product moved along mechanized conveyors, but the workstations along the line were manually operated) and machining transfer lines (beginning around 1924).

3.1.1.2. Programmable automation, the production equipment is designed with the capability to change the operations to accommodate different product configurations. Some of the typical features that characterize programmable automation include:

* High investment in general-purpose equipment

* Low production rates relative to fixed automation

* Flexibility to deal with changes in product configuration

* Most suitable for batch production

Automated production systems that are programmable are used in low and medium-volume production. The parts or products are typically made in batches. To produce each new batch of a different product, the system must be reprogrammed with the set of machine instructions that correspond to the new product. Examples of programmable automation include numerically controlled machine tools (first prototype demonstrated in 1952) and industrial robots (initial applications around 1961), although the technology has its roots in the Jacquard loom (1801).

3.1.1.3. Flexible automation is an extension of programmable automation. The concept of flexible automation has developed only over the last 15 or 20 years, and the principles are still evolving. A flexible automated system is one that is capable of producing a variety of products (or parts) with virtually no time lost for changeovers from one product to the next. There is no production time lost while reprogramming the system and altering the physical setup (tooling, fixtures, machine settings). Consequently, the system can produce various combinations and schedules of products, instead of requiring that they be made in separate batches. The features of flexible automation can be summarized as follows:

* High investment for a custom-engineered system

* Continuous production of variable mixtures of products

* Medium production rates

* Flexibility to deal with product design variations

The essential features that distinguish flexible automation from programmable automation are:

a. The capacity to change part programs with no lost production time.

b. The capability to change over the physical setup, again with no lost production time.

The above features allow the automated production system to continue production without the downtime between batches that is characteristic of programmable automation. Changing the part programs is generally accomplished by preparing the programs off-line on a computer system and electronically transmitting the programs to the automated production system. Therefore, the time required to do the programming for the next job does not interrupt production on the current job. Advances in computer systems technology are largely responsible for this programming capability in flexible automation. Changing the physical setup between parts is accomplished by making the changeover off-line and then moving it into place simultaneously as the next part comes into position for processing.

Although it is the intention of the thesis to emphasize the use of computers in manufacturing, but first, how to prepare a process plan manually is shown.

CHAPTER 4

4. Manual Process Planning

There are variations in the level of detail found in route sheets among different companies and industries. In the one extreme, process planning is accomplished by releasing the part print to the production shop with the instructions "make to drawing." Most firms provide a more detail list of steps describing each operation and identifying each work center. The traditional approach to process planning has been to manually examine an engineering drawing and develop a detailed process plan based upon knowledge of process and machine capabilities, tooling, materials, and shop practices. The process planner, given a specific material, must endeavor to build a list of processes for converting the given raw material into its desired final geometry. In any case, it is traditionally the task of the manufacturing engineers or industrial engineers in an organization to write these process plans for the new part designs to be produced by the shop. Manual process planning leans very heavily upon the judgement and experience of the planner. It is the manufacturing engineer's responsibility to determine an optimal routing for each new part design and requires well trained, experienced personnel who are well versed in shop floor practices. However, individual engineers each have their own opinions about what constitutes the best routing and would always exhibit the personal preferences and prejudice of the individual.

For a model shop, where all the machinists are highly skilled in several machines and most parts produced are one of a kind, the process plan is usually nothing but a list of workstation routes. The remaining detail is left to the machinists. In the case where a part is produced in an entirely automated transfer line, the process plan contains a breakdown of every detail. Machines, tools, fixtures, and dies are designed to execute the required operations. However, these are extreme cases; in most machine shop, small-batch production is the norm.

In order to prepare a process plan, a process planner has to have the following knowledge:

* Ability to interpret an engineering drawing

* Familiarity with manufacturing processes and practice

* Familiarity with tooling and fixtures

* Know what resources are available in the shop

* Know how to use reference books, such as machinability data handbooks

* Ability to do computations on machining time and cost

* Familiarity with the raw materials

* Know the relative costs of processes, toolings, and raw materials.

Objective/Criterion in selectivity of a Process Plan, the following are some steps that have to be taken:

a. Economic Criteria

a.1. Minimum processing time

a.2. Minimum material cost

b. Study the overall shape of the part. Use this information to classify the part and determine the type of workstation needed.

c. Thoroughly study the drawing. Try to identify all manufacturing features and notes.

d. Determine the best raw material shape to use if raw stock is not given.

e. Identify datum surfaces. Use information on datum surfaces to determine the setups.

f. Select machine for each setup.

g. Determine the rough sequence of operations necessary to create all the features for each setup.

h. Sequence the operations determined in the previous step. Check whether there is any interference or dependency between operations. Use this information to modify the sequence or operations.

i. Select tools for each operation. Try to use the same tool for several operations if possible. Keep in mind the trade-off on tool-change time and estimated machining time.

j. Select or design fixtures for each setup.

k. Select or design jigs for each setup.

l. Evaluate the plan generated thus far and make necessary modifications.

m. Select cutting parameters for each operation.

n. Prepare the final process-plan document.

One has to keep in mind that during each of these steps, decisions are made based on an evaluation of many factors. For example, tool selection is based on the feature to be created and other related features as well as the machine to be used. Operations selection depends on the features to be created and the capability of the machine selected. Machine selection is normally determined by the operations required.

The following are some of the disadvantages of manual process planning:

a. The required experience for a manual planner requires years of hard work to accumulate.

b. The experience thus developed would not apply to new processes.

c. Experience gathered would represent approximate data only.

The disadvantages of this method has left to investigate the automation of the process planning function.

CHAPTER 5

5. Computer-Aided Process Planning

Because of the problems encountered with manual process planning, attempts have been made in recent years to capture the logic, judgement and experience required for this important function and incorporate them into computer programs. The use of computer resources to aid the process planner in a systematic determination of proper methods to be used in the production processes is called CAPP. These systems manage the storage, retrieval, distribution, and maintenance of the process plan library. They have been extended to include the logic used by process planners in choosing alternate process methods. Early attempts to automate process planning consisted primarily of building computer-assisted systems for report generation, storage, and retrieval of plans. A database system with a standard form editor is what many early systems encompassed. Formatting of plans was performed automatically by the system. Process planners simply filled in the details. The storage and retrieval of plans are based on part number, part name, or project ID. When used effectively, these systems can save up to 40% of a process planner's time.

Perhaps the best known automated process planning system is the CAM-I automated process-planning system, CAPP (CAM-I stands for Computer-Aided Manufacturing--International, a

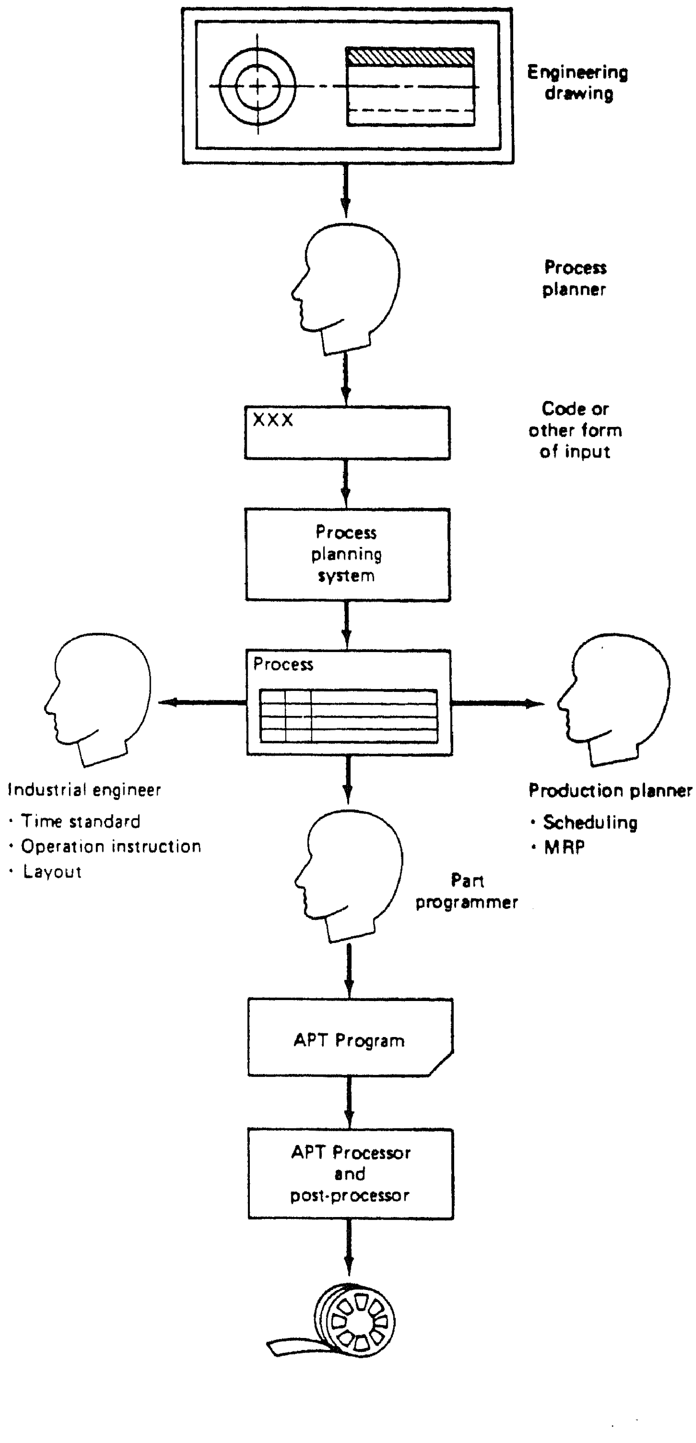


FIGURE 10. Typical Process-Planning System.

nonprofit industrial research organization). In CAPP, previously prepared process plans are stored in a database. When a new component is planned, a process plan for a similar component is retrieved and subsequently modified by a process planner to satisfy special requirements. The technique involved is called Group-Technology (GT) based variant planning. The typical organization of a variant process-planning system is shown in Figure 10. During the last decade or so, there has been much interest in automating the task of process planning of CAPP systems. The shop-trained people who are familiar with the details of machining and other processes are gradually retiring, and these people will be unavailable in the future to do process planning. An alternative way of accomplishing this function is needed, and CAPP systems are providing this alternative. Recent developments in Computer-Aided Process Planning have been focused on eliminating the process planner from the entire planning function. Computer-Aided Process Planning can reduce some of the decision making required during a process planning. It has the following advantages:

a. It can reduce the skill required of a planner.

b. It can reduce the process-planning time.

c. It can reduce both process-planning and manufacturing costs.

d. It can create more consistent plans.

e. It can produce more accurate plans.

f. It can increase productivity.

Process planning is the critical bridge between design and manufacturing, Figure 11 illustrates this situation. Design information can be translated into manufacturing language only through process planning. Today, both automated design, Computer-Aided Design (CAD) and manufacturing, Computer-Aided Manufacturing (CAM) have been implemented. Integrating, or bridging, these functions require automated process planning.

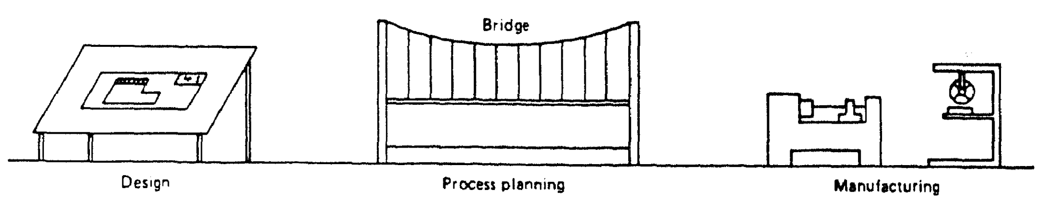


FIGURE 11. Process-Planning Bridges Design And Manufacturing.

5.1. Benefits of CAPP

Among the benefits derived from computer-aided process planning, some of them are the following:

a. Process Rationalization and Standardization: Automated process planning leads to more logical and consistent plans than when process planning is done completely manually. Standard plans tend to result in lower manufacturing costs and higher product quality.

b. Increased Productivity of Process Planners: The systematic approach and the availability of standard process plans in the data files permit more work to be accomplished by the process planners. One system was reported to increase productivity by 600%.

c. Reduced Lead Time for Process Planning: Process planners working with the CAPP system can provide route sheets in a shorter lead time compared to manual preparation.

d. Improved Legibility: Computer-prepared route sheets are neater and easier to read than manually prepared route sheets.

f. Incorporation of Other Application Programs: The CAPP program can be interfaced with other application programs, such as cost estimating, work standards, and others.

Modern CAPP systems are focused around two technologies. These technologies are called:

5.2. Retrieval CAPP systems

5.3. Generative CAPP systems

5.2. Retrieval CAPP systems

The Retrieval-type computer-aided process planning systems also called Variant CAPP systems requires a catalog of standard process plans to be stored in a computer systems. The part being planned is classified and coded using the principles of Group Technology (GT). GT approach is required for parts classification and coding system to organize the computer files and to permit efficient retrieval of the appropriate process plan. The variant approach uses library retrieval procedures to find standard plans for similar components. A process plan that can be used by a family of components is called a standard plan. A standard plan is stored permanently in the database with a family code number as its key. The standard plans are created manually by process planners. The standard process plans are based on current part routings in use in the factory, or on an ideal plan that is prepared for each family. There is no limitation to the detail that a standard plan can contain. However, it must contain at least a sequence of fabrication steps or operations. For some new parts, when a standard plan is retrieved, a certain degree of modification is usually necessary in order to use the plan on a new component. The development of the data base of these process plans requires substantial effort.

The retrieval method and the logic in variant systems are predicated on the grouping of parts into families. Common manufacturing methods can then be identified for each family. Such common manufacturing methods are represented by standard plans.

The mechanism of standard-plan retrieval is based on part families. A family is represented by a family matrix that includes all possible members. In use, a retrieval CAPP system operates as illustrated in Figure 12. The user begins by deriving the GT code number for the component for which the process plan is to be determined. With this code number, a search is made of the part family file to determine if a standard route sheet exists for the given part code. If the file contains a process plan for the part, it is retrieved and displayed for the user. The standard process plan is examined to determine whether any modifications are necessary. It might be that the new part has the same code

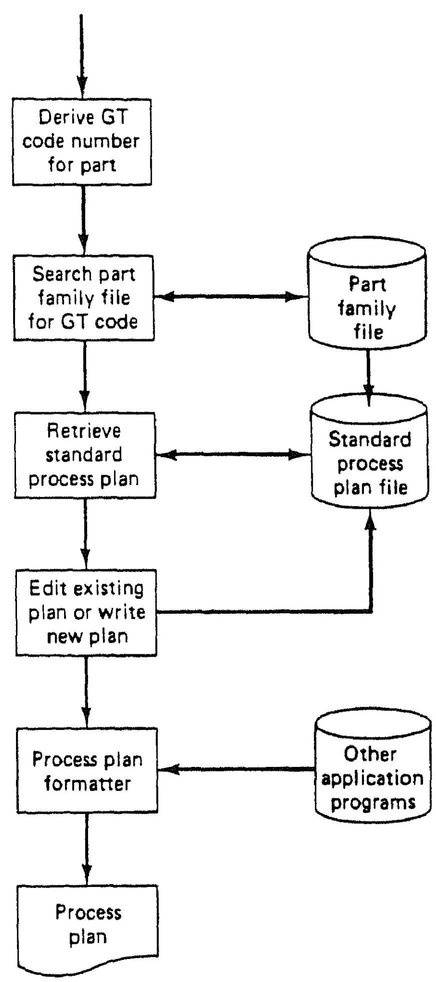


FIGURE 12. General Procedure For Using One Of The Retrieval Computer-Aided Process Planning Systems.

number, there are minor differences in the processes to make the part. The user edits the standard plan accordingly. It is this capacity to alter an existing process plan that gives the retrieval system its other name: Variant CAPP system.

If the file does not contain a standard process plan for the given code number, the user may search the computer file for a similar or related code number for which a standard route sheet does exist. Either by editing an existing process plan, or by starting from scratch, the user writes the route for the new part. This route sheet becomes the standard process plan for the new part code number. In general, variant process-planning systems have two operational stages:

5.2.1. The Preparatory Stage

5.2.2. A Production Stage

5.2.1. The Preparatory Stage

The Preparatory work is required when a company first starts implementing a variant system. During the preparatory stage, existing components are coded, classified, and subsequently grouped into families. The first step is to choose an appropriate coding system. The coding system must cover the entire spectrum of parts produced in the shop. It must be unambiguous and easy to understand. The special features exist on the parts must be clearly identified by the coding system. An existing coding system may be adopted and then modifications can be made for the specific shop. The coding of existing components can be a tedious task. Before it can be done, a thorough study of the inventory of drawings and process plans has to be completed so that an orderly coding task can be conducted. The personnel involved in coding must be trained. They must have the precise understanding of the coding system. They must generate the identical code for the same component when they work independently. Inconsistent coding of components results in redundant and erroneous data in the database.

After coding is completed, part families can be formed. A family matrix is then constructed for each part family. Due to the large number of components involved, a computer should be used to help construct family matrices. The next step is to prepare standard process plans for part families. The preparatory stage is a labor-intensive process. It requires a tremendous amount of effort. Whatever is prepared for shop A can be used only for shop A. The system structure and software can be used by other shops, but the database must be prepared uniquely by and for each shop.

5.2.2. The Production Stage

The production stage occurs when the system is ready for use. New components now can be planned. An incoming component is first coded.

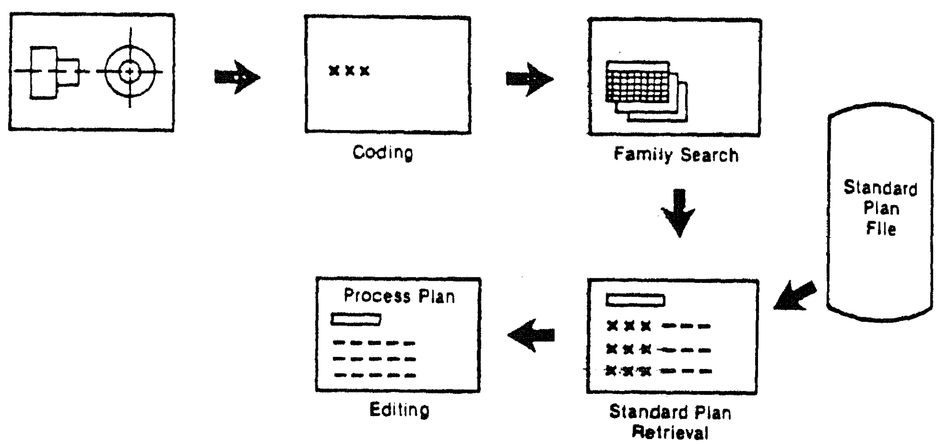


FIGURE 13. Production Stage.

The code is then input to a part-family search routine to find the family to which the component belongs. The family number is then used to retrieve a standard plan. The human planner may modify the standard plan to satisfy the component design. For a frequently produced part, it might be desirable to perform the search by direct code matching. In this case, a process plan (not a standard plan) for an existing part is retrieved. Figure 13 shows the flow of the production stage. Some other functions, such as parameter selection and standard time calculations, can also be added to make the system more complete.

5.2.3. Database Structure

Because of the large amount of information, database systems play an important role in variant process planning. A database is no more than a group of cross-referenced data files. The database contains all the necessary information for an application and can be accessed to by several different programs for specific applications. There are three approaches to construct a database:

a. **Hierarchical**

b. **Network**

c. **Relational**

Although the concept and structure of these approaches are very different, they can serve the same purpose.

For commercial programming, there are several available database management systems, such as CODASYL, ORACLE, RBASE, UNIFY, dBASE-V, Quatro Pro, and Lotus 1-2-3. These systems are high-level languages for database construction and manipulation. Of course, a database can always be written using procedural languages such as COBOL, FORTRAN, and C. No matter what approach and language are used, the basic structure of the database is the same.

5.2.4. Search Procedure

Once the preparatory stage has been completed, the variant planning system is ready for production. The basic idea of the variant system is to retrieve process plans for similar components. The search for a process plan is based on the search of a part family to which the component belongs. When the part family is found, the associated standard plan can be easily retrieved.

5.2.5. Plan Editing and Parameter Selection

Before a process plan can be released to the shop, some modification of the standard plan is necessary, and process parameters must be added to the plan. There are two types of plan editing: one is the editing of the standard plan itself in the database, and the other is the editing of the plan for the component. Editing a standard plan implies that a permanent change in the stored plan be made. This editing must be handled very carefully because the effectiveness of a standard plan affects the process plans generated for the entire family of components. Aside from the technical considerations of file maintenance, the structure of the database must be flexible enough for expansion and additions and deletions of data records.

Editing a process plan for a component requires the same expertise as editing a standard plan. However, it is a temporary change and , therefore, does not affect any other component in the family. During the editing process, the standard plan has to be modified to suit the specific needs of the given component. Some operations or entire records have to be removed and others must be changed. Additional operations may also be required to satisfy the design. A text editor is usually used at this stage.

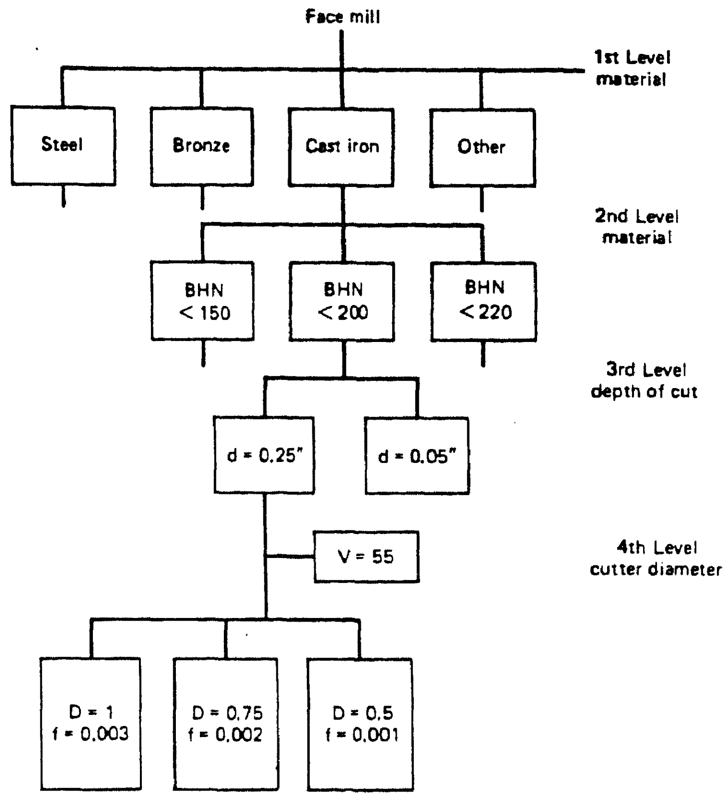


FIGURE 14. Process-Parameter File.

A complete process plan includes not only operations, but also process parameters.

Figure 14 shows the structure for the parameter file. Data in the file are linked so that we can go through the tree to find the feed and speed for an operation.

The parameter file can be integrated into variant process plan to select process parameters automatically. Information such as depth of cut and cutter diameter can be retrieved directly from the process plan for each operation.

5.3. Generative CAPP systems

Generative CAPP systems is a process planning system that synthesizes process information in order to process plans for parts automatically. It uses an automatic, computerized system consisting of decision logic, formulae, technology algorithms, and geometry-based data to make decisions required for a process plan. The computer would employ a set of algorithms to progress through the various technical and logical decisions toward a final plan for manufacturing. Inputs to the system would include a comprehensive description of the workpart. This may involve the use of some form of part code number to summarize the workpart data, but it does not involve the retrieval of existing standard plans. Instead, the generative CAPP system synthesizes the design of the optimum process sequence, based on an analysis of part geometry, material, and other factors which would influence manufacturing decisions. The problem of designing a generative CAPP system is considered part of the field of expert systems, a branch of artificial intelligence. An expert system is a computer program that draws on the organized expertise, it is capable of solving complex problems that normally require one or more human experts who has years of education and experience. Process planning fits within the scope of that definition.

There are several ingredients required in fully generative process planning system.

a. The first ingredient is the technical knowledge of manufacturing and the logic that is used by successful process planners must be captured and coded into a computer program. In an expert system as it would be applied to process planning, the knowledge and logic of the human process planners is incorporated into a so-called "knowledge base". The generative CAPP system would then use that knowledge base to solve process planning problems (i.e., create route sheet).

b. The second ingredient in generative process planning is a computer-compatible description of the part to be produced. This description contains all of the pertinent data and information needed to plan the process sequence. Two possible ways of providing this description are:

b.1. The geometric model of the part that is developed on a CAD system during product design.

b.2. A GT code number of the part that defines the part features in significant detail.

c. The third ingredient in a generative CAPP system is the capability to apply the process knowledge and planning logic contained in the knowledge base to a given part description. In other words, the CAPP system uses its knowledge base to solve a specific problem--planning the process for a new part. The problem-solving procedure is referred to as the "inference engine" in the terminology of expert system. By using its knowledge base and inference engine, the CAPP system synthesizes a new process plan from scratch for each new part it is presented.

Other planning functions, such as machine selection, tool selection, and process optimization, can also be automated using generative planning techniques.

The generative planning techniques/approach has the following advantages:

a. It can generate consistent process plans rapidly.

b. New components can be planned as easily as existent components.

c. It can potentially be interfaced with an automated manufacturing facility to provide detailed and up-to-date control information.

Decisions on process selection, process sequencing, etc., are all made by the system. However, transforming component data and decision rules into a computer-readable format is still a major obstacle to be overcome before generative planning systems become operational. Successful implementation of this approach requires the following key developments:

a. Process-planning knowledge must be identified and captured.

b. The part to be produced must be clearly and precisely defined in a computer-readable format (e.g., three-dimensional model and GT code).

c. The captured process-planning knowledge and the part description data must be incorporated into a unified manufacturing database.

Today the term "Generative Process Planning' is often loosely used. Systems with built-in decision logic are often called generative process-planning systems. The decision logic consists of the unusual ability to check some conditional requirements of the component and select a process. Some systems have decision logic to select several "canned" process-plan fragments and combine them into a single process plan. However, no matter what kind of decision logic is used and how extensively it is used, the system is usually categorized as a generative system.

Generative process plan design requires long term investments of people, machinery, and time. Generative process planning techniques can use forward planning or backward planning. Forward planning involves modifying the workpiece until it attains the features required by the finished product. Backward planning involves starting with the finished product, and filling it to the shape of the unmachined workpiece. Each machining process is considered as a filling process.

There are several generative process-planning systems used in the industry (e.g., AUTOPLAN OF METCUT [Vogel, 1979; Vogel and Adlard, 1981], and CPPP of United Technology [Dunn and Mann, 1978]).

Ideally, a generative process-planning system is a turnkey system with all the decision logic coded in the software. The system possesses all the necessary information for process planning; therefore, no preparatory stage is required. This is not always the case, however. In order to generate a more universal process-planning system, variables such as process limitation, process capabilities, and process costs must be defined prior to the production stage. Systems such as CPPP (Kotler, 1980a, 1980b) require user-supplied decision logic (process models) for each component family. A wide range of methods have been and can be used for generative process planning.

5.3.1. Forward and Backward Planning

In variant process planning, process plans are retrieved from a database. A direction for the planning procedure does not exist because plans are simply linked to a code. However, in generative process planning, when process plans are generated, the system must define an initial state in order to reach the final state (goal). The path taken (initial-to-final or final-to-initial) represents the sequence of processes. For example, the initial state is the raw material (workpiece) and the final state is the

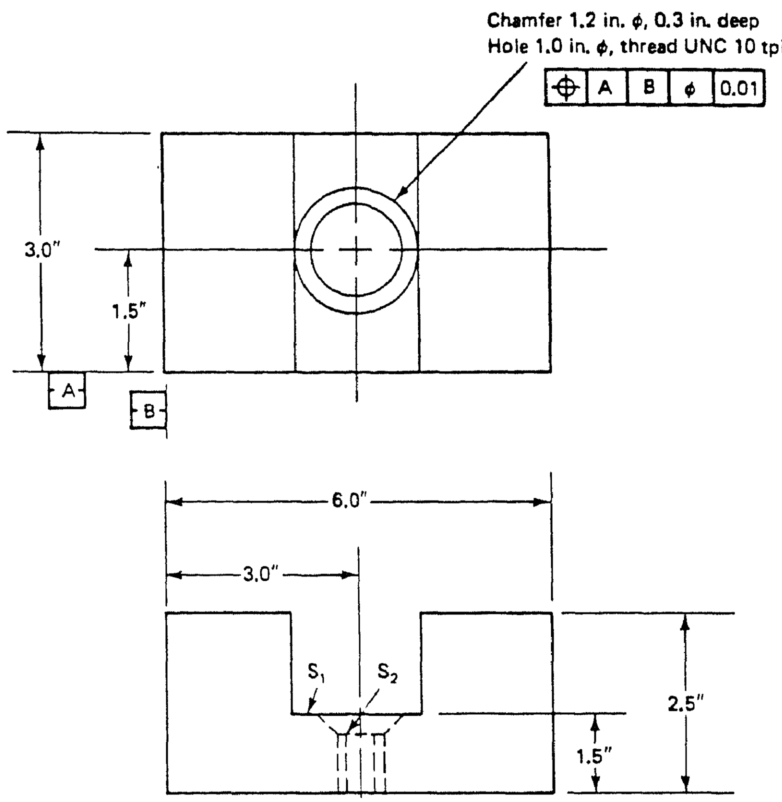


FIGURE 15. Component To Be Planned: Final Design.

component design. If the planner works on modifying the raw workpiece until it takes on the final design qualities, the type of component is shown in Figure 15. The raw workpiece is a 6.0-in. X 3.0 in. X 2.5 in. block. Starting from the top surface S1, using forward planning. A milling process is used to create the final dimension for S1. After S1 has been planned, S2 can be planned. For S2, a chamfering process is first selected, and then the hole is drilled and tapped. The progression begins from the raw material and proceeds to the finished requirements.

Backward planning uses a reverse procedure. Assuming that we have a finished component, the goal is to fill out to the unmachined workpiece shape. Each machining process is considered a filling process. A drilling process can fill a hole; a reaming process can fill a thin wall (cylinder); and so on. When applied to the example component, the bottom most surface is planned first. A tapping process is selected (this reduces the threaded surface to a smaller-diameter hole with a rough surface finish). Drilling is then selected to fill the hole, and so forth, until we finally obtain the block.

Forward and backward planning may seem similar, however, they affect the programming of the system significantly. Planning each process can be characterized by a precondition of the surface to be machined and a post-condition of the machining (its results). For forward planning, we must know the successor surface before we select a process, because the post-condition of the first process becomes the precondition for the second process. For example, when we selected drilling for the threaded hole, we knew that the thread was going to be cut. Therefore, we rough drilled the hole using a smaller drill. Otherwise, we might have chosen a larger drill and no thread could be produced. Backward planning eliminates this conditioning problem since it begins with the final surface forms and processes are selected to satisfy the initial requirements. The transient surface (intermediate surface) produced by a filling process is the worst precondition a machining process can accept (i.e., depth of cut left for finish milling. etc.). Any filling process that can satisfy the transient surface can be selected as the successor process.

In forward planning, the objective surface must always be maintained even though several operations must be taken to guarantee the result. On the other hand, backward planning starts with the final requirements (which helps to select the predecessor process) and searches for the initial condition or something less accurate (which is easy to satisfy).

5.3.2. The Input Format

The input format of a process-planning system affects the ease with which a system can be used and the capability of the system. A system using a very long special description language as its input is more difficult to use. The translation from the original design (either an engineering drawing or a CAD model) to a specific input format may be tedious and difficult to automate. In this case, it is probably easier and faster to plan a component manually than to prepare the input. However, such input can provide more complete information about a component, and more planning functions can be accomplished using the input. This does not imply that a system using a long and special descriptive language always provides more planning functions.

Many different input formats have been used in process-planning systems. Although no/few systems use the same input format, we can categorize the input for these systems into the following classes:

a. Code

b. Description Language

c. CAD Models

a. Codes

As discussed in the variant approach, GT codes can be used as input for variant process-planning systems. Some generative systems such as APPAS (Wysk, 1977) and GENPLAN (Tulkoff, 1981) also use part codes as input. Codes used in generative systems are more detailed and sometimes mix code digits with explicitly defined parameter values. Since a code is concise, it is easy to manipulate. When process capabilities are represented by a code, a simple search through the process capability to match the component code will return the desired process. In order to determine the process sequence, a code for the entire component is appropriate because it provides global information.

b. Description Language

Specially designed part-description language can provide detailed information for process-planning systems. A language can be designed to provide all of the information required for the necessary functions of a process-planning system. The format can be designed such that functions can easily accomplish their task from the information provided.

AUTAP system (Eversheim et al., 1980) uses a language similar to a solid modeling language. A component is described by the union of some primitives and modifiers. Figure 16 shows the description of a rotational component.

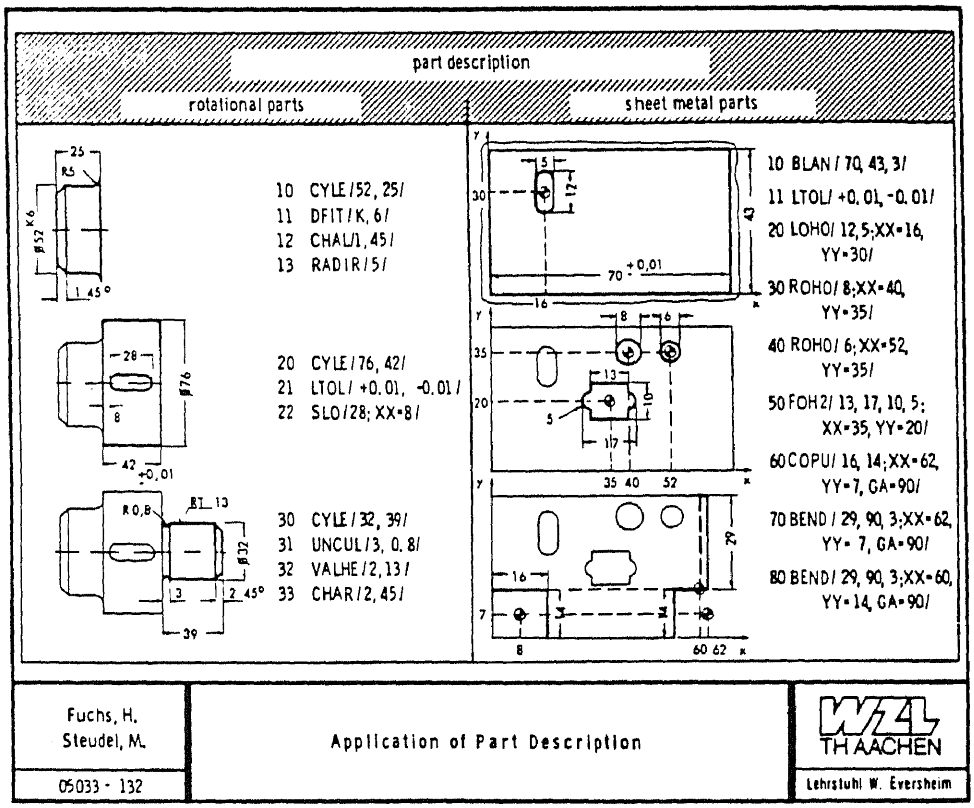
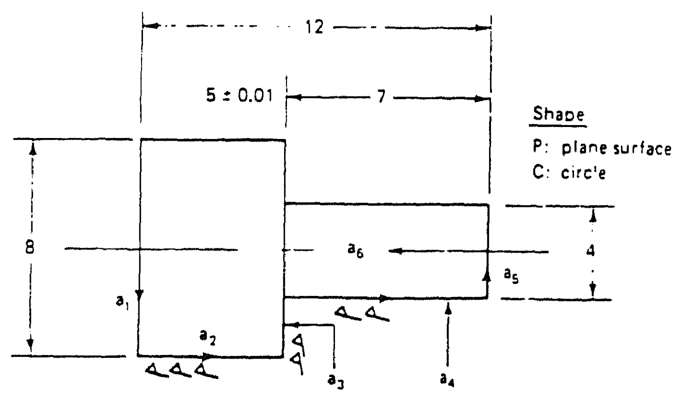


FIGURE 16. AUTAP Data Input (Eversheim et al., 1980).

CYCLE (cylinder), CHAL (chamfer left), CHAR (chamfer right), UNCUL (undercut), and RADIR (radius right-curved chamfer) are primitives and modifiers. A process planner can model the component using the language. Materials, processes, machine selection, and time estimates can be selected by the system using the input model. Although reasonably complex components can be modeled, this language lacks a complete set of Boolean operators, and modeling a complex component may be difficult. The process sequence is also affected directly by the sequence with which a component is modeled. Although the system models a component from left to right, it does not reduce the number of possible models for a component to a single description.



	Shape code	Dimensions		S.F.	Tol.	Adjacent shape
a_1	P	0	4		± 0.01	a_6
a_2	C	5	0	P P P		a_1
a_3	P	0	−2	P P		a_2
a_4	C	7	0	P P		a_3
a_5	P	0	2			a_4
a_6	•	−12	0			a_5

FIGURE 17. CIMS/DEC Component Modeling.

Another system, CIMS/PRO, developed by Iwata et al. (1980), uses an input language called CIMS/DEC (kakino et al., 1977). In the CIMS/DEC systems, component shape is modeled by sweeping (translation or rotation) to generate surfaces, Figure 17. In CIMS/PRO, a pattern-recognition module automatically identifies machined surfaces, such as planes, cylinders, threaded shafts, holes, and grooves. Twenty six different types of machined surfaces can be characterized. Tool approach can also be determined (every machined surface has a set of approach directions predefined). This input language can model both rotational (by rotation sweep) and boxlike (by translation sweep) components. It is most powerful for modeling rotational components without axial-shape elements. However, if the component is very complex, it is difficult to model.

GARI (Descotte and Latombe, 1981) is an Artificial Intelligence (AI) problem solver. It is one of the earliest attempts to use AI in process planning.

A component can be described by some system words such as diameter and surface finish, Figure 18.

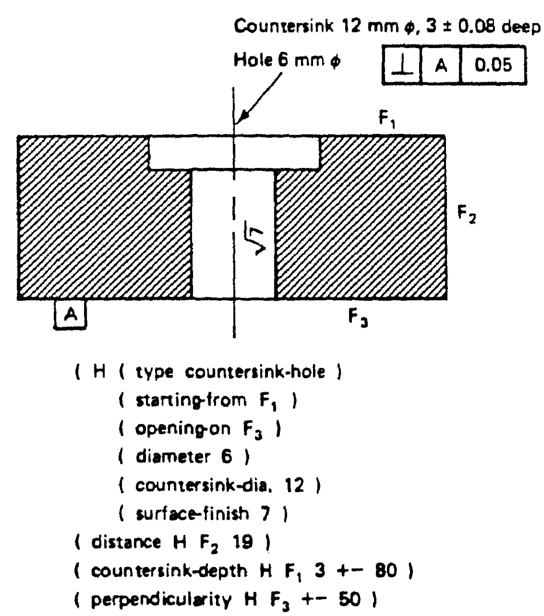


FIGURE 18. GIRI Component Modeling.

Rules can then be applied to determine the processes and machines needed to produce a part. The knowledge base (where process and machine capabilities are stored) uses the same set of system words; therefore, decisions can be reached by searching the knowledge in order to satisfy the input description. The input description must, however, be prepared by a human operator. For a complex component, the translation of the original design to this input language can be tedious and difficult.

There are many other systems such as CPPP [Kotler, 1980a, 1980b] and AUTOTECH [Tempelhof, 1980] that use their own special-shape code, dimensions, and technological data. Although description languages can provide complete information for process-planning functions, the main problem (the difficulty to generate the original design automatically) is still unresolved. The next class of input format is aimed at eliminating this problem.

c. CAD Models

Since a design can be modeled effectively in a CAD system, using a CAD model as input to a process-planning system can eliminate the human effort of translating a design into a code or other descriptive form. The increased use of CAD in industry further points to the benefits of using CAD models for process-planning data input.

A CAD model contains all the detailed information about a design. It can provide information for all the planning functions. However, an algorithm to identify a general machined surface from a CAD model does not currently exist. Additional code is needed to specify the machined surface shape from the raw material shape.

CADCAM (Chang and Wysk, 1981) uses a CAD model for its input. Several other systems (AUTOPLAN [Vogel and Adlard, 1981] and GENPLANO also use a CAD database interactively for tool and fixture selection. There is tremendous potential for using CAD data for process-planning input. However, substantial work has to be done.

5.3.3. Decision Logic

In a generative process-planning system, the system decision logic is the core of the software and directs the flow of program control. The decision logic determines how a process or processes are selected. The major function of the decision logic is to match the process capabilities with the design specification. Process capabilities can be described by "IF....THEN...." expressions. Such expressions can be translated into logical statement in a computer program. Perhaps the most efficient way to translate these expressions is to code process-capability expressions directly into a computer language. Information in handbooks or process boundary tables can be easily translated using a high-level computer language.

Several methods can be used to describe the decision structure of process planning. The knowledge-representation methods are related directly to the decision logic in these systems. The static data are the representation and the dynamic use of the data becomes the decision logic. The following are the list of the decision logic as applied to process-planning systems. The list is by no means complete. However, this classification forms a handy framework for the discussion.

5.3.3.1. Decision Trees

5.3.3.2 Decision Tables

Decision trees and decision tables are ways of describing or specifying the various actions (decisions) associated with combinations of (input) conditions. Both decision trees and decision tables have been employed to help systematize decision making. With the advent of the digital computer, increased use of these tools have occurred. Both can be described by the following example. The activity deals with a weekend decision that can be described as follows:

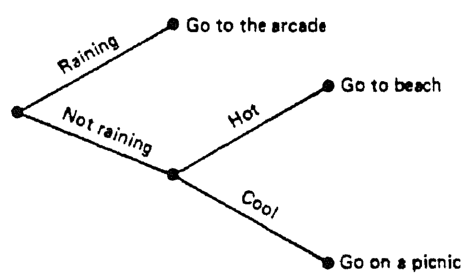
"If it is raining, I will go to the arcade and play video games."

"If it is not raining and hot, I will go to the beach."

"If it is not raining and cool, I will go on a picnic with a friend."

Raining	T	F	F
Hot		T	F
Go to the arcade	X		
Go to beach		X	
Go to picnic			X

(a) Decision table



(b) Decision tree

FIGURE 19. Decision Table And Decision-Tree Representation.

This plan can also be represented by either a decision table or a decision tree, Figure 19.

In a decision table, T implies true to a specified condition, F implies false, and a blank entry implies that we do not care. Only when all conditions in a decision table column are met is the marked action taken.

In a decision tree, the conditions are specified on the branches of the tree. When all of the branches leading to an action hold true, the action at the terminal point is taken. This system can be used represent the decision for the weekend decision plan given, as well as knowledge base for process planning.

Both decision trees and decision tables are tools to assist in decision making. Since decision rules must cover all possible situations, they must be well thought out before such a tool can be used for process planning. For a given set of decision rules, one can list the conditions and actions, as shown in the example, and then translate the actions into decision tree or decision table form. Whatever can be represented by a decision tree can also be represented by decision table. The primary difference is the ease and elegance of presentation and programming when a computer is used.

5.3.3.1. Decision Trees

A decision tree is a natural way to represent process information. Conditions (IF) are set on the branches of the tree and predetermined actions can be found at the junction of each branch.

A decision tree can be implemented as either;

a. computer code or
b. presented as data.

When a decision tree is implemented in computer code, the tree can be directly translated into a program flow chart. The root is the start node, Figure 19, and each branch is a decision node. Each branch has a decision statement (a true condition and a false condition). At each junction, an action block is included for the true condition. For a false condition, another branch might be taken or the process might be directed to the end of the logic block. When the false condition includes another branch, these two branches are said to branch from an OR node. When the false condition goes directly to the end of an action block (which is rooted from the same decision statement), the current branch and the following branch are part of the same AND node. A decision statement can be a predicate or a mathematical expression.

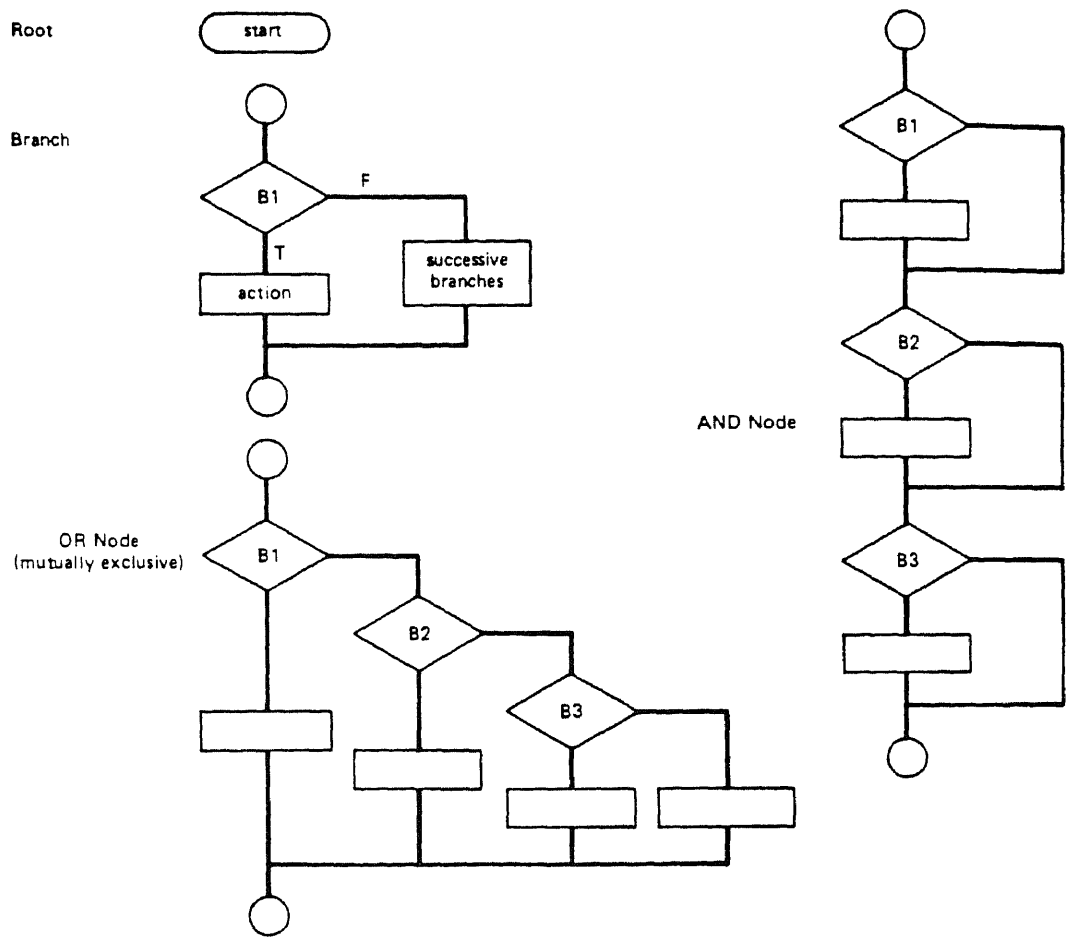


FIGURE 20. Structured Flow Chart Corresponds To A Decision-Tree.

Figure 20 shows a sample decision tree and its flow-chart representation. It can be written in a programming language (pseudo-code) as follows:

```
;root
          if E1 then do N1 enddo
          else if E7 then do A5 enddo
          else endif endif stop

;node N1
;
procedure N1
if E2 then do N2 enddo
          else if E3 then do A4 enddo
          else endif endif return

;node N2
;
procedure N2
if E4 then do A1 enddo
          else if E5 then do A2 enddo
          else if E6 then do A3 enddo
          else endif endif endif return.
```

This Language (or interpreter) format allows for the easy construction of decision trees that are frequently used in generative process-planning systems. Similar language formats using FORTRAN, Pascal, C, and so on, have been developed for general-purpose algorithms. APPAS (Wysk, 1977) is a typical example of decision-tree logic used for process planning. Although the approach is easy to implement, system-expansion work can be difficult, especially for a programmer other than the original author of the system.

When implementing a decision tree in data form, another program (system program) is required to interpret the data and achieve the decision-tree flow. This approach is more difficult to develop (for the system program). However, once the system program has been developed, the implementation and system maintenance are significantly lessened. Again, however, it can be extremely difficult to add a function that was not originally included in the system program. There are many methods that can be used to design such a system program. A simple example is presented to demonstrated a basic structure that one can use.

We call this example system DCTREE, DCTREE uses a query procedure to obtain design information and then print the final conclusions. In DCTREE, there are three major components:

a. The decision-tree data

b. A compiler

c. A system run-time module

Part (a) is supplied by a user who translates a decision tree from graph form into DCTREE input-language format. The DCTREE compiler compiles the input and saves it in a computer-usuable format. Finally, the system run-time module uses the compiled decision tree to generate questions, make decisions, and print out conclusions.

We first look at the input language. These are two parts of each input. Expression definition and Tree-structure definition, Figure 21. In the expression definition, each expression is preceded by an expression identifier (ID). Each ID must be unique. An expression with an ID initial of Q or A (query or action) is simply stored in a buffer. Other expressions are compiled as condition expressions. A condition expression (such as &1 < or = 0.002) uses postfix notation and stack operations. A variable (&1) causes the run-time module to input a real number and store it on a stack. Therefore, these expressions can be compiled as a simple code instead of using a simple constant or a variable datum.

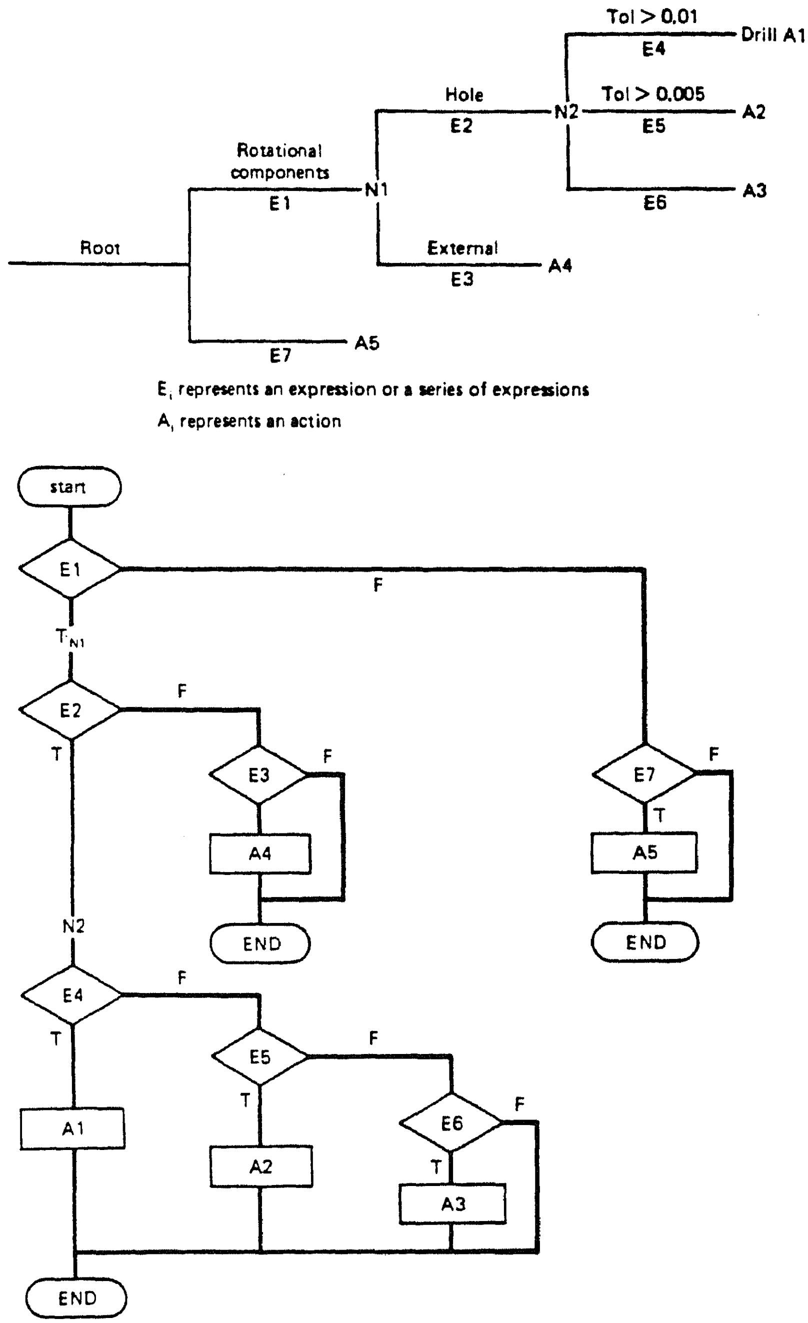


FIGURE 21. Implementation Of A Decision-Tree In A Program.

The tree structure is represented by expression IDs and pointers. For instance, an arrow (--->) represents "point to." The syntax is given below:

$$E_{no} \text{ ---> } \begin{bmatrix} \text{AND} \\ \text{OR} \end{bmatrix} (E_{n1}, E_{n2}, \ldots\ldots E_{nm}) \; \Big| \; A_i$$

E_{no} = root branch (source)

E_{ni} = expression number (destination action)

A_i = execution action

| = either E's or A's, but not both

During compilation, each is assigned an address in the tree structure file. E_{ni}'s in parentheses are substituted by pointers. A_i's are marked with negative values to indicate their actions. Figure 22 shows how a decision tree can be represented by DCTREE.

The system run-time module performs the I/O and decision making.

	Expression definition
Q1	Hole diameter ?
E1	$\&1 > 0.0$
Q2	True position ?
E2	$\&1 \leq 0.002$
E3	$(\&1 \leq 0.01)$. AND . $(0.002 < \&1)$
E4	$\&1 > 0.01$
Q5	Tolerance ?
E5	$\&1 \leq 0.002$
E6	$(\&1 \leq 0.01)$. AND . $(0.002 < \&1)$
E7	$0.01 < \&1$
A1	Rapid travel out, true position = 0.01
A2	Finish bore, true position = 0.02
A3	Finish bore, tolerance = 0.01
A4	Semifinish bore, tolerance = 0.02
A5	Drill, diameter = 0
Q8	Slot ?
Q9	Internal thread ?
E8	&1
E9	&1
A6	Mill
A7	Tap

Tree structure definition

0 → OR (E1 E8 E9)
E1 → AND (E2 E3 E4)
E2 → A1
E3 → A2
E4 → AND (E5 E6 E7)
E5 → A3
E6 → A4
E7 → A5
E8 → A6
E9 → A7

Decision tree

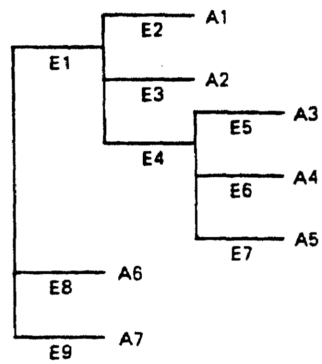


FIGURE 22. Input To DCTREE.

A decision tree used in the design of SORICH CAPP system is illustrated in the Figure 23.

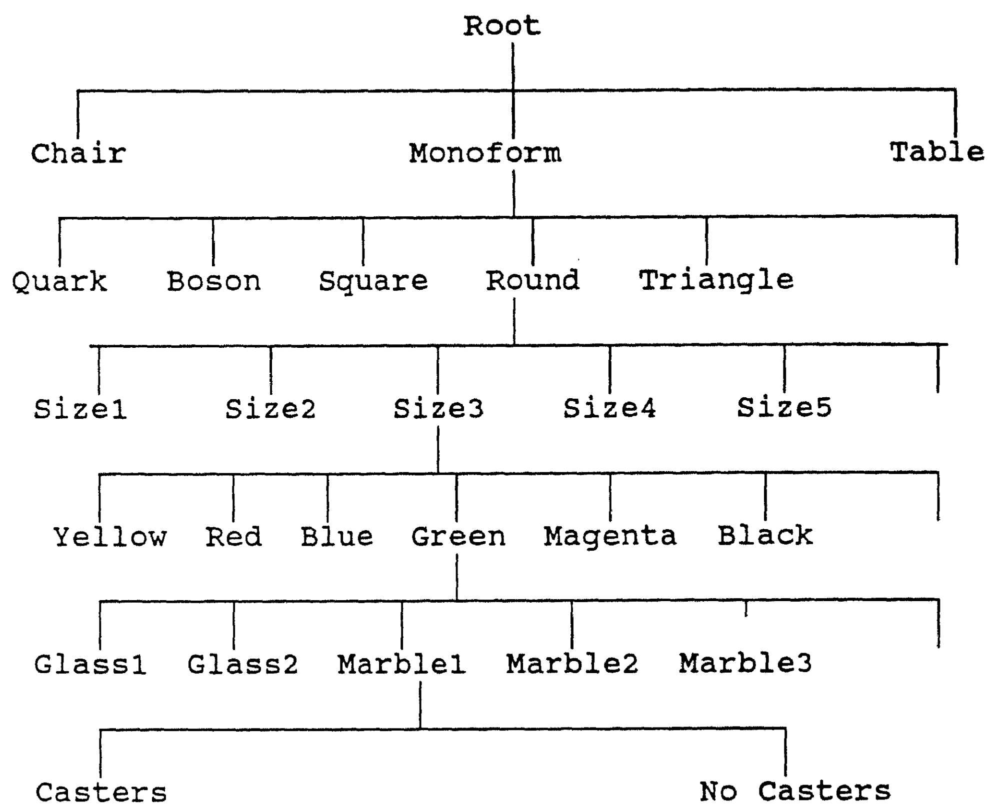


FIGURE 23. SORICH Decision-Tree

(Product Oriented).

5.3.3.2. Decision Tables

Decision tables have long been used to present complex engineering data. Decision tables can also be easily implemented on a computer. Using decision tables for process planning, however, normally requires a special preprocessor program or computer language to implement the table and control the operation of the table. Such software is generally called a decision-table language. A decision-table language consists of:

a. A base language
b. A decision table
c. An outer language

A base language is the foundation of a decision-table language. For example, FORTAB of the RAND Corporation and S/360 DLT of the IBM Corporation use FORTRAN as their base language (McDaniel, 1970). DETAB/65 (Silberg, 1971) of SIGPLAN of the ACM (Association of Computing Machinery) uses COBOL as its base language. A base language is extended to include statements that can describe a decision table in a more easily implemented manner. A preprocessor is occasionally written to translate the decision-table language program into its base-language program.

The decision table is the most essential part of a decision-table-language program. It is represented in its original table format. Decision-table techniques can be used to simplify and/or parse a complex table.

The third element (outer language) is used to control the decision table. Decision table does not contain any input/output or control statements; therefore, it is not a complete program. An outer language can eliminate this void.

5.4. SORICH CAPP System Designed for Thesis

The CAPP system produced using the Quattro Pro software is a cross between Retrieval Process Planning and Generative Process Planning. It uses the concept from generative process planning of asking questions of the user that help describe the product to be manufactured. Upon answering these questions a code number is generated. This code number is then used to retrieve the appropriate files. This is where the variant type concepts are used. If the system does not find a file that matches exactly the code number which was generated, it allows the user to retrieve the closest code number file and make the necessary changes. The exact way in which the software performs these tasks will be described in the section on software use.

CHAPTER 6

6. Group Technology in CAPP

Since the beginning of human culture, people have tried to apply reason to their actions. One important way to apply reason is to relate similar things. Biologists classify items into genus and species. We relate to such things as mammals, marsupials, batrachians, amphibians, fish, mollusks, crustaceans, birds, reptiles, worms, insects, and so on. A chicken is a bird with degenerated wings. Tigers, jaguars, and domestic cats are all members of a single family.

The same concept applied to natural phenomena can also be applied to fabrication and information phenomena. When a vast amount of information has to be kept and ordered, a taxonomy is normally employed. Librarians use taxonomies to classify books in libraries. Similarly, in manufacturing, thousands of items are produced yearly. When one looks at the parts that construct the product, the number is exceptionally large. Each part has a different shape, size, and function. However, when one looks closely, one may again find similarities among components in Figure 24, a dowel and a small shaft may be very similar in appearance but different in function. Spur gears of different sizes need the same manufacturing processes and vary only in size.

Therefore, it appears that manufactured components can be classified into families similarly to biological families or

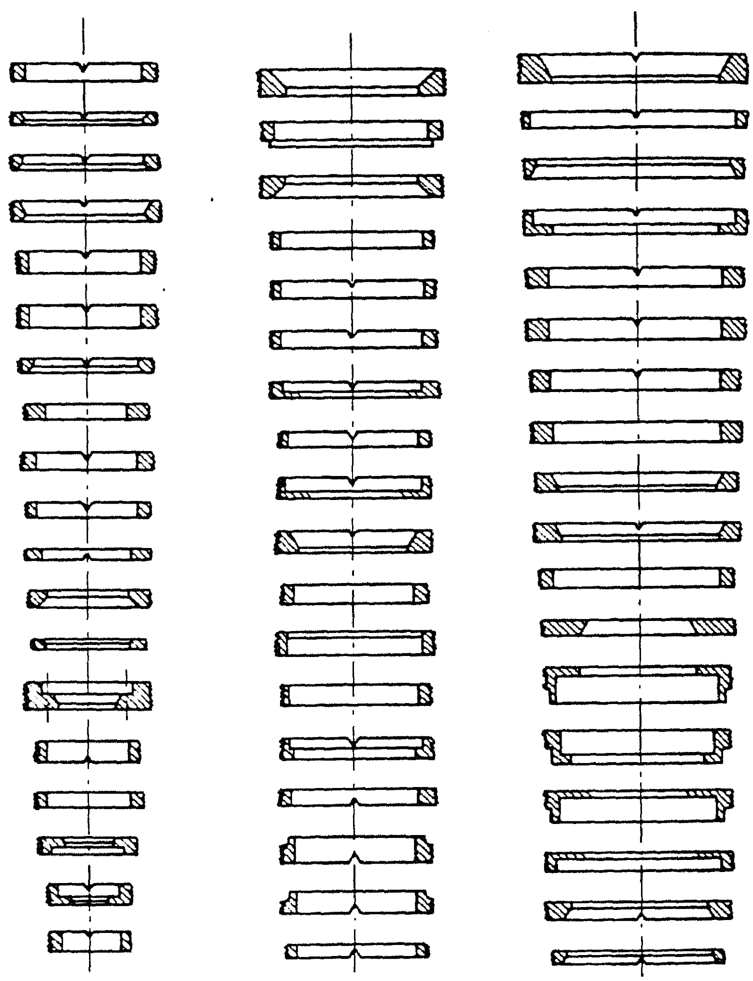


FIGURE 24. Design Family.

library taxonomies. Parts classified into families similarly to biological families or library taxonomies. Parts classified and grouped into families produce a much more tractable database for management.

Group Technology (GT) is a manufacturing philosophy in which similar parts are identified and grouped together to take advantage of their similarities in manufacturing and design. Although this simple concept has been in existence for a long time, it was not until 1958 that S. P. Mitrofanov, a Russian engineer, formalized the concept in his book "The Scientific Principles of Group Technology". Group Technology (GT) has been defined (Solaja, 1973) as follows:

"Group Technology is the realization that many problems are similar, and that by grouping similar problems, a single solution can be found to a set of problems thus saving and effort."

Although the definition is quite broad, one usually relates group technology only to production applications. In production systems, group technology can be applied in different areas. For component design, it is clear that many components have a similar shape (Figure 24). Similar components, therefore, can be grouped into design families and a new design can be created by simply modifying an existing component design from the same family. By using this concept, composite components can be identified. Composite components are parts that embody all the features of a design family or design subfamily. An example is shown in Figure 25. Components in the family can be identified from features of the composite components.

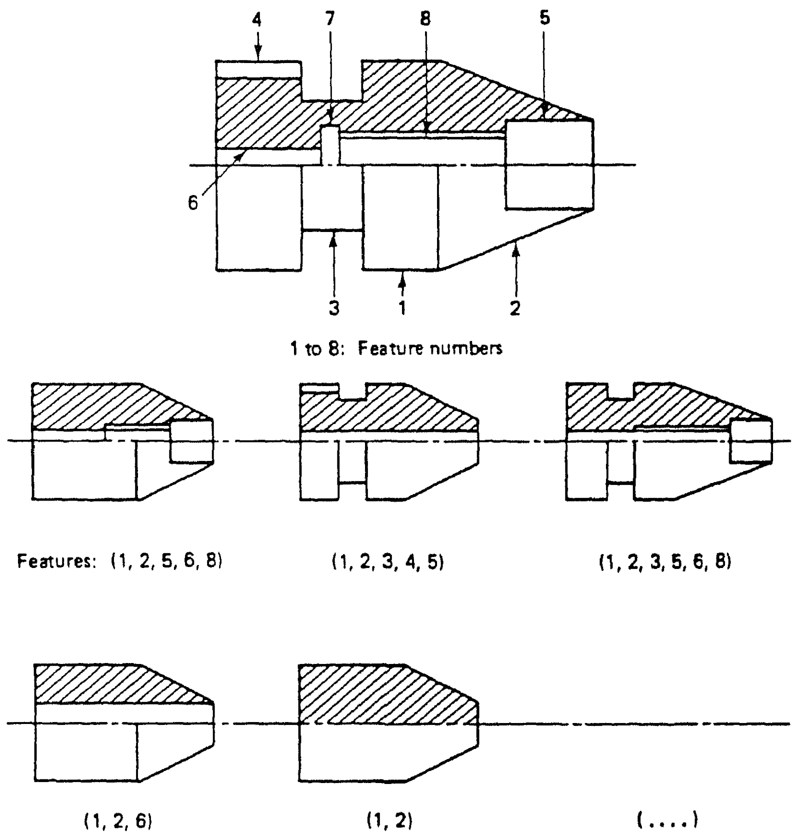


FIGURE 25. Composite Components.

For manufacturing purposes, GT represents a greater importance than simply a design philosophy. Components that are not similar in shape may still require similar manufacturing processes. For example, in Figure 10.3, most components have different shapes and functions, but all require internal boring, face milling, hole drilling, and so on. Therefore, it can be concluded that the components in the figure are similar. The set of similar components can be called a production family. From this, process-planning work can be facilitated. Since similar processes are required for all family members, a machine cell can be built to manufacture the family. This makes production planning and control much easier, since only similar components are considered for each cell. Such a cell oriented layout is called a group-technology layout or cellular layout.

6.1. Part-Family Formation

One of the major benefits derived from GT applications is part family formation for efficient work flow. Efficient work flow can result from grouping machines logically so that material handling and setup can be minimized. Parts can frequently be grouped so that the same tooling and fixtures can be used. When this occurs, a major reduction in setup

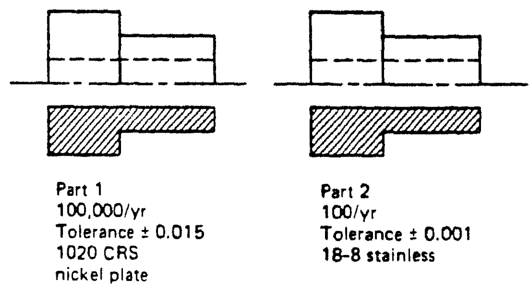


FIGURE 26. Two Parts Of Identified Shape And Size But Different Manufacturing Requirements.

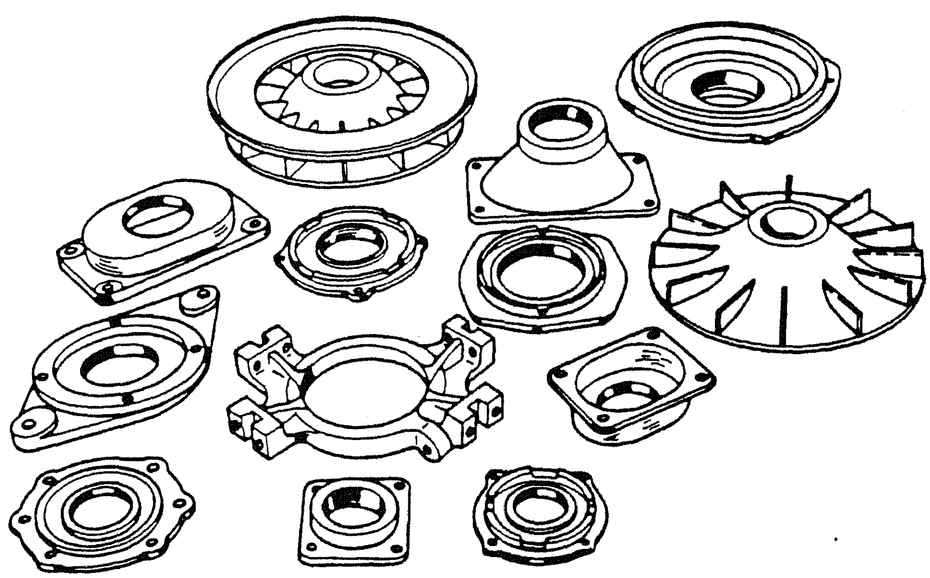


FIGURE 27. Thirteen Parts With Similar Manufacturing Process Requirements But Different Design Attributes.

results. Machines can also be grouped so that the amount of handling between machining operations also can be minimized. A part family is a collection of parts which are similar either because of geometric shape and size or because similar processing steps are required in their manufacturing. The parts within a family are different, but their similarities are close enough to merit their identification as members of the part family. Figure 26 and 27 show two part families. The two parts shown in Figure 26 are similar from a design viewpoint but quite different in terms of manufacturing. The parts shown in Figure 27 might constitute a part family in manufacturing, but their geometry characteristic do not permit them to be grouped as a design part family.

The part family concept is central to design retrieval systems and most current computer-aided process planning schemes. Another important manufacturing advantage derived from grouping workparts into families can be explained with reference to Figures 28 and 29. Figure 28 shows a process-type (functional) layout for batch production in a machine shop. It is one of the most common machine layouts used in industry As can be seen in the figure, machines are laid out with respect to type. Mills, lathes, drills, and grinders are clustered so that they reside in a single department. This layout is efficient in the allocation of tooling and fixturing within departments. It also allows a supervisor to

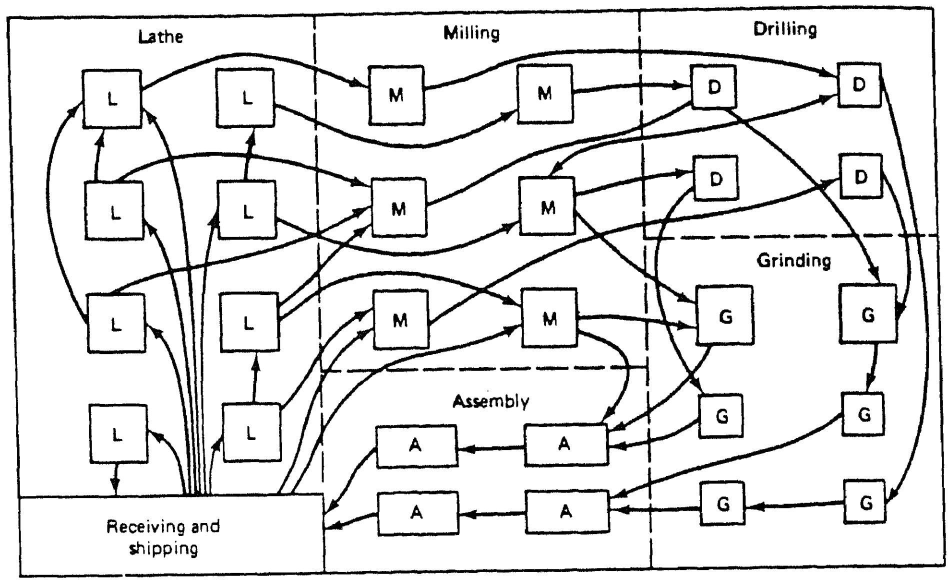


FIGURE 28. Process-Type Layout.

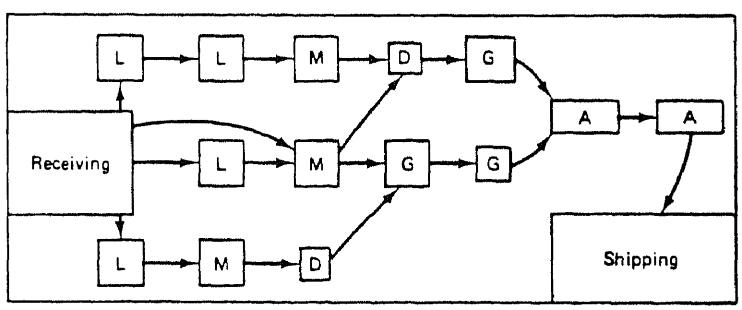


FIGURE 29. Group Technology Layout.

accumulate a significant amount of information concerning a limited set of operations within the department. Unfortunately, functional layout also requires that the product flows throughout the entire system in a somewhat random manner (as can be seen in the figure). Material handling can be significantly reduced if parts requiring similar operations are clustered into a "part family", and the machines required to produce the part family are organized into a cell. This process is known as cellular layout and is illustrated in Figure 29. As can be seen from the figure, the flow of product through the system is much more direct and material handling can be reduced significantly.

The basis of cellular, or GT layout is part-family formation. Family formation is based on parts production or more specifically, their manufacturing features. Components requiring similar processes are grouped into the same family.

There is no rigid rule that can be applied to form families. Users must set their own definitions of what a family is or should be. A general rule in forming the family is that all the parts must be related. For production-flow analysis, all parts in a family must require similar routings. A user may want to put only those parts having exactly the same routing sequence into a family.

The biggest single obstacle in changing over to group technology from a traditional production shop is the problem of grouping parts into families. Before grouping can start, information concerning the design and processing of all the existing components must be collected from existing part and processing files. There are three general methods for solving this problem. All three methods are time consuming and involve the analysis of much data by properly trained personnel. The three methods are:

6.2. Visual Methods

6.3. Production Flow Analysis (PFA)

6.4. Parts Classification and Coding System

6.2. Visual Methods

The visual inspection method is the least sophisticated and least expensive method. It involves the classification of parts into families by looking at either the physical parts or photographs and arranging them into similar groupings. It is obvious that when we have many components, visual inspection will be difficult to use. This method is generally considered to be the least accurate of the three.

TABLE 4. PARTIAL COMPONENT PROCESS SUMMARY

Component	Code	Processes	OP Code Sequence		
A-112	1110	SAW01,	LATHE02,	GRIND05,	INSP06
A-115	6514	MILL02,	DRL01,	INSP03	
A-120	2110	SAW01,	LATHE02,	GRIND05,	INSP06
A-123	2010	SAW01,	LATHE01,	INSP06	
A-131	2110	SAW01,	LATHE02,	INSP06	
A-212	7605	MILL05,	INSP03		
A-230	6604	MILL05,	INSP03		
A-432	2120	SAW01,	LATHE02,	INSP06	
A-451	2130	SAW01,	LATHE02,	INSP06	
A-510	7654	MILL05,	DRL01,	GRIND06,	INSP06
A-511					
A-511					
A-512					
A-550					
A-556					
B-105					
B-107					
B-108					
B-109					
B-115					
B-116					
B-117					
B-118					
B-119					
B-120					

6.3. Production Flow Analysis (PFA)

For example, if we have a total of 24 components in our minishop. After coding them, we can obtain a summary in Table 4. We can easily group parts into families with the help of computerized methods such as production-flow analysis.

	A-112	A-115	A-120	A-123	A-131	A-212	A-230	A-432	A-451	A-510
SAW01	/		/	/	/			/	/	
LATHE01				/						
LATHE02	/		/		/			/	/	
DRL01		/								/
MILL02		/								
MILL05						/	/			/
GRIND05	/		/							
GRIND06	/		X							/
INSP03		/				/	/			
INSP06	/		/	/	/			/	/	/

FIGURE 30. PFA Matrix.

Production Flow Analysis (PFA) was introduced by J. L. Burbidge to solve the family-formation problem for manufacturing cell design (Burbidge, 1971, 1975). Many researchers have subsequently developed algorithms to solve the problem. In PFA, a large matrix (incidence matrix) is constructed. Each row represents an OP code, and each column in the matrix represents a component, Figure 30. We can define the matrix as M_{ij}, where i designates the OP codes, and j designates components (M_{ij} = 1 if component j has OP code i; otherwise, M_{ij} = 0). The objective of PFA is to bring together those components that need the same or a similar set of OP codes in clusters.

6.3.1. Rank Order Cluster Algorithm

J. R. King (1979) presented a rank-order cluster algorithm that is quite simple. We use his method to show how component families can be determined in our shop. King's algorithm can be stated as follows:

Step 1. Calculate total weight of each column as:

$$w_j = \sum_i 2^i M_{ij}$$

Step 2. If w^j is in ascending order, go to step 3, or rearrange in ascending order.

Spep 3. Calculate total weight of each row as:

$$w_i = \sum_j 2^j M_{ij}$$

Step 4. If in ascending order stop, else rearrange and return to step 1.

The rank-order clustering algorithm sorts the matrix into a diagonal block structure. The diagonal blocks are not always mutually exclusive. Final judgement has to be made by the user.

One of the major drawbacks of applying this algorithm is the need of storing the binary word. The word length is max(n, m), where n is the number of machines, and m is the number of components. For a moderate problem with 50 machines and 2000 components, it is impossible to calculate the weights before sorting. For the machine weights, 50 words of 2000 bits are needed. A word of 2000 bits requires 250 bytes. To overcome this problem, direct comparison of elements, either row or column, can be used. A digit-by-digit comparison is performed, beginning from the most significant digit. Each row or column of the matrix is treated as a binary number; no weight is ever calculated. Unfortunately, this procedure has a computational complexity of a cubic order, namely,

	A112	A115	A120	A123	A131	A212	A230	A432	A451	A510	
SO1	/		/	/	/			/	/		2
LO1				/							4
LO2	/		/		/			/	/		8
DO1		/								/	16
MO2		/									32
MO5						/	/			/	64
GO5	/										128
GO6										/	256
W_j	138	48	10	6	10	64	64	10	10	336	
Rank	9	6	2	1	3	7	8	4	5	10	

	A123	A120	A131	A432	A451	A115	A212	A230	A112	A510	W_i	Rank
SO1	/	/	/	/	/				/		574	4
LO1	/										2	1
LO2		/	/	/	/				/		602	5
DO1						/					1088	7
MO2						/					64	2
MO5							/	/		/	1408	8
GO5									/		512	3
GO6										/	1024	6
	2	4	8	16	32	64	128	256	512	1024		

	A123	A120	A131	A432	A451	A115	A212	A230	A112	A510	
LO1	/										2
MO2						/					4
GO5									/		8
SO1	/	/	/	/	/				/		16
LO2		/	/	/	/				/		32
GO6										/	64
DO1						/				/	128
MO5							/	/		/	256
W_j	18	48	48	48	48	132	256	256	56	448	
Rank	1	2	3	4	5	7	8	9	6	10	

	A123	A120	A131	A432	A451	A112	A115	A212	A230	A510	W_i	Rank
LO1	/										2	1
MO2							/				128	5
GOF						/					64	2
SO1	/	/	/	/	/	/					126	4
LO2		/	/	/	/	/					124	3
GO6										/	1024	6
DO1							/			/	1152	7
MO5								/	/	/	1792	8
	2	4	8	16	32	64	128	256	512	1024		

	A123	A120	A131	A432	A451	A112	A115	A212	A230	A510		
LO1	/										2	Part Family #1
GO5						/					4	
LO2		/	/	/	/	/					8	
SO1		/	/	/	/	/					16	
MO2							/				32	Part Family #2
GO6										/	64	
DO1							/			/	128	
MO5								/	/	/	256	
W_j	18	24	24	24	24	26	160	256	256	510		

Stop

FIGURE 31. Rank-Order Cluster Algorithm.

O[ij(i+j)] (King and Nakornchai, 1982), where i and j are the number of rows and columns, respectively.

Figure 31 shows the procedure of rearranging the PFA matrix in Figure 30. Note that the last two rows in Table 3 were not used in Figure 31 in order to simplify the example.

After we obtain the final matrix, we can determine (arbitrarily) that components A123, A120, A131, A432, A451, and A112 form a family that needs SAW01, LATHE01, LATHE02, and GRIND05. A115, A212, A230, and A510 form the second family.

6.4. Coding and Classification

Parts classification and coding systems are divided into one of the following four general categories:

a. CAD--Systems based on part design attributes

b. CAM--Systems based on part production attributes

c. CAB--Systems based on part business attributes

d. Combinations

* CAD/CAM--Systems based on both design and production attributes

* CIM--Systems based on business, design and production attributes

Coding is a process of establishing symbols to be used for meaningful communication. Classification is a separate process in which items are separated into groups based on the existence or absence of characteristic attributes. Coding can be used for classification purposes, and classification requirements must be considered during the construction of coding scheme. Therefore, coding and classification are closely related.

Before a coding scheme can be constructed, a survey of all component features must be completed and then code values can be assigned to the features. The selection of relevant features depends on the application of the coding scheme. For example, tolerance is not important for design retrieval, therefore, it is not feature in a design-oriented coding system. However, in a manufacturing oriented coding system, tolerance is indeed an important feature. Because the code structure affects its length, the accessibility and the expendability of a code (and the related database) is

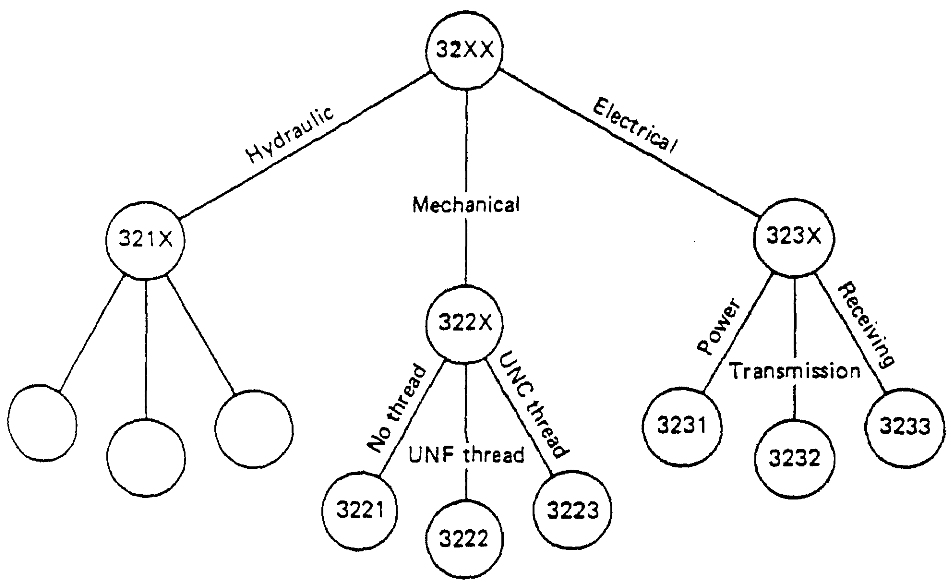


FIGURE 32. Hierarchical Structure.

of importance. There are three different types of code structure in GT coding system:

6.4.1. Hierarchical

6.4.2. Chain (matrix)

6.4.3. Hybrid.

6.4.1. Hierarchical

A Hierarchical structure is also called a monocode. In a monocode, each code number is qualified by the preceding characters. For example, in Figure 32, fourth digit indicates threaded or not threaded for a 322X family. One advantage of a hierarchical structure is that it can represent a large amount of information with very few code positions. A drawback is the potential complexity of the coding system. Hierarchical codes are difficult to develop because of all the branches in the hierarchy that must be defined.

A hierarchical type structure is being used in our custom designed CAPP system. In this way we can represent more code numbers with Quattro-Pro's limited eight digit file name allowance.

6.4.2. Chain (matrix)

A chain structure is called a polycode. Every digit in the code position represents a distinct bit of information, regardless of the previous digit.

TABLE 5. CHAIN STRUCTURE

Digit position	1	2	3	4
Class of feature	External shape	Internal shape	Holes	
Possible value				
1	Shape 1	Shape 1	Axial	
2	Shape 2	Shape 2	Cross	
3	Shape 3	Shape 3	Axial and cross	
4				

In Table 5, a chain-structured coding scheme is presented. A 2 in the third position always means a cross hole no matter what numbers are given to positions 1 and 2. Chain codes are compact and are much easier to construct and use. The major drawback is that they cannot be as detailed as hierarchical structures with the same number of coding digits.

6.4.3. Hybrid.

The third type of structure, the hybrid structure, is a mixture of the hierarchical and chain structures Figure 33. Most existing coding system use a hybrid structure to obtain

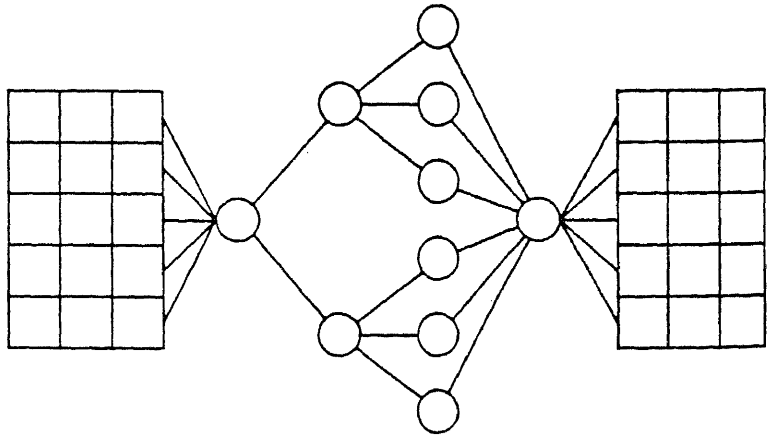


FIGURE 33. Hybrid Structure.

the advantages of both structures. A good example is the widely used Optiz (1970) code Figure 34. There are more than 100 GT coding systems used in industry today. The structure selected is based primarily on the application.

The physical coding of a component can be shown best by example. In Figure 35, a rotational component is coded using the Optiz system. By going through each code position, the resulting code becomes 01112. This code represents this component and all others with similar shape and diameter. A definition of the specific Optiz code attributes is given later in Figure 36.

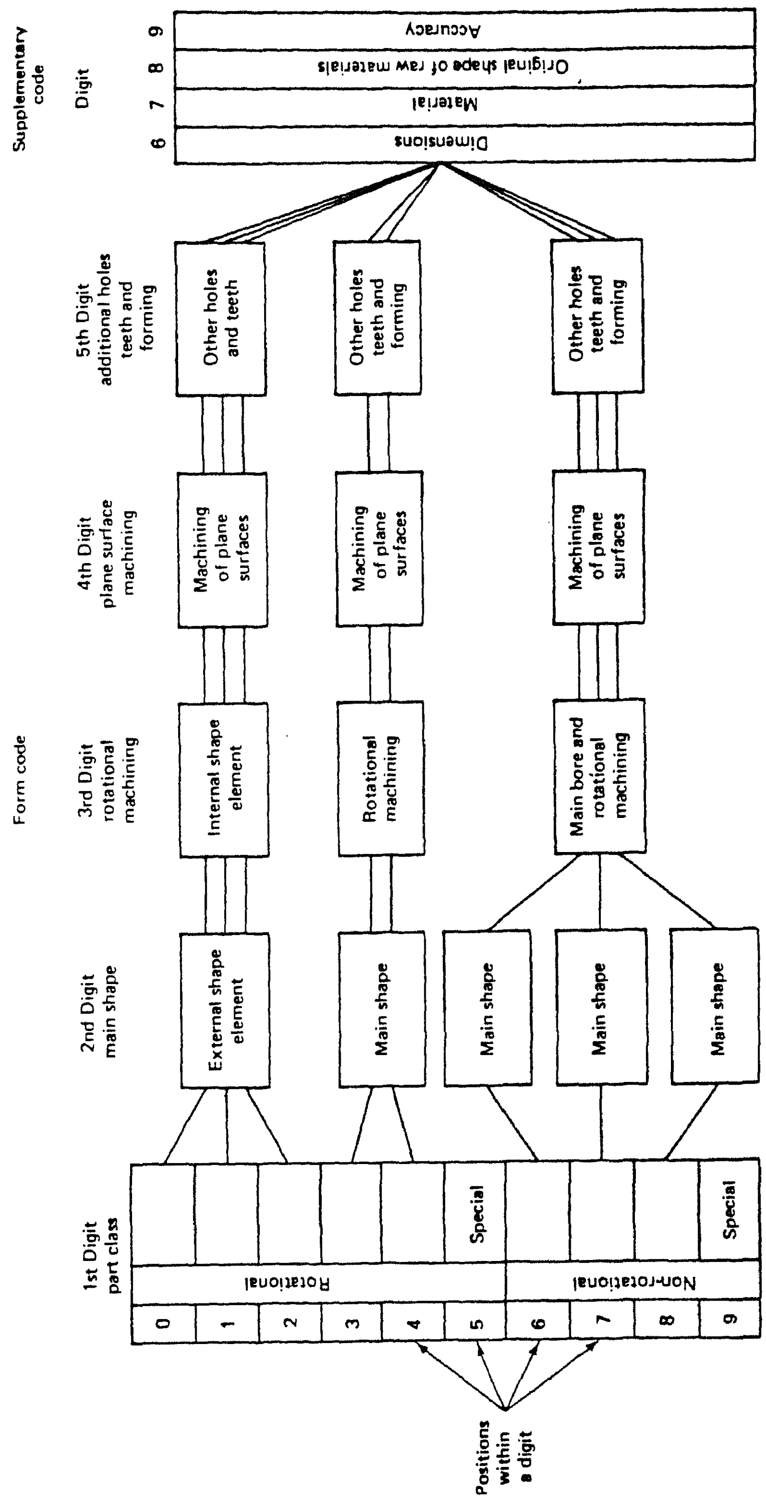


FIGURE 34. Opitz Coding And Classification System.

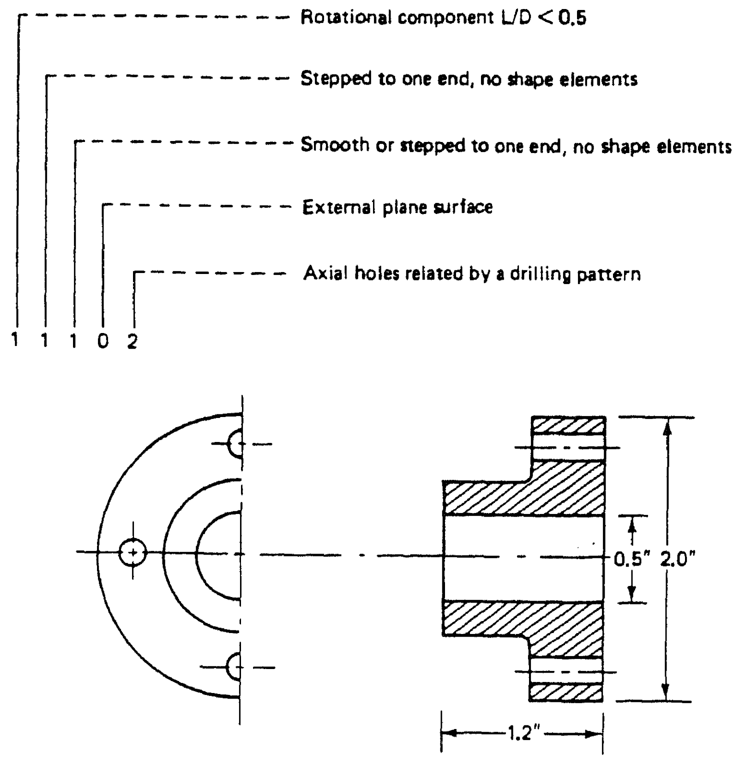


FIGURE 35. Rotational Component (H. Opitz, A Classification System To Describe Workpieces).

6.5. Group-Technology Coding System

Both drafting and geometric modeling are detailed representations of an engineering design. They provide detailed information concerning the component to be made and are essential in conveying the design for manufacturing. However, as with many decision-making processes, too much information may make the decision more difficult. For example, when reading a magazine or journal, one seldom begins by reading each sentence on successive pages. Instead, you peruse the table of contents first in order to locate interesting technical information. By doing so, candidate articles can be located more quickly. However, the title may not convey all of the necessary information. Therefore, the reader would typically scan the abstract. The abstract is summary that represents the article without great detail. Reading the abstract of an article normally provides you with the insight to continue reading. Parts lists corresponding to a table of contents have been around for a long time, and although it is impractical to write abstracts for CAD model, A similar concept can be applied using coding.

Group Technology (GT) is an appropriate tool for this purpose. Coding, a GT technique, can be used to model a component without all the detail. When constructing a coding system for a component's representation, there are several factors to be considered. They include:

a. The population of components (i.e., rotational, prismatic, deep drawn, sheet metal, etc.)

b. The detail the code should represent

c. The code structure: chain, hierarchical, hybrid

d. The digital representation(i.e., binary, octal, decimal, alphanumeric, hexadecimal, etc.)

The population of component contributes to the variety of shapes. For example, the population in the United States include virtually all races that exist on earth. In a sense, it is necessary to distinguish race, hair color, eye color, and so on. However, in a nation such as China or Japan, it is not worthwhile to record skin color, hair color, and so on, because these items are virtually invariant. In component coding, it is also true that only those features that vary have to be included in the code . When designing or using a coding scheme, two properties must hold true: the code must be:

a. **Unambiguous and**

b. **Complete.**

The code has to be concise. If a 100-digit code and a 10-digit code can both represent components in population space completely and unambiguously , the 10-digit code is normally necessary. For example, the basic Opitz code shown in Figure 32 uses five digits to describe the shape. Five digits can represent 10^5 combinations. With this set, it is not possible to show a large amount of detail of a component. Some codes are significantly longer, for example, the KK-3 of Japan (Japan Society, 1980), which has 21 digits and contains multiple digits for single features, and MICLASS of TNO (Houtzeel and Schilperoot, 1970), which has a 12-digit code. For some computer-aided process planning systems (e.g., APPAS [Wysk, 1977]), a detailed surface code is used instead of a code for the entire component. The decision on how much detail the code should represent depends solely on the application. The selection of a code structure again depends on the application.

The last consideration in coding-system construction is the code digits. Several positional alternatives can be selected (from binary to alphanumeric). However, this selection yields different precisions for the different schemes. For example, an N-digit code with different coding features yields the following combinations of code:

Binary	**2**	**(0,1)**
Octal	**8**	**(0,1,....,7)**
Decimal	**10**	**(0,1, ,9)**
Hexadecimal	**16**	**(0,1,...., 9,A,....,F)**
Alphanumeric	**(26 + 10)**	**(0,1,...., 9,A,.....Z)**

Although alphanumeric systems are the most compact (the same amount of information can be represented by fewer digits), the difficulty of handling both numerical and alphabetical characters makes alphanumerics less attractive.

6.5.1. The Opitz System

The Opitz coding system (Opitz, 1970) was developed by H. Opitz of the Aachen Tech University in the West Germany. The code uses a mixed code structure. However, except for the first digit, it resembles a chain structure more closely. Table 6. The Opitz code is a 9-digits code, it is probably the best known coding system. First 5-digits describe the geometry (form/shape) and rest 4-digits are supplementary code. The geometric code can represent parts of the following variety:

* Rotational
* Flat
* Long
* Cubic

A dimension ratio is further used in classifying the geometry:

TABLE 6. FORM CODE (DIGITS 1-5) FOR ROTATIONAL PARTS IN THE OPITZ SYSTEM. PART CLASSES 0, 1, AND 2

	Digit 1		Digit 2			Digit 3		Digit 4	Digit 5	
	Part class		External shape, external shape elements			Internal shape, internal shape elements		Plane-surface machining	Auxiliary holes and gear teeth	
0	Rotational parts	$L/D \leq 0.5$			Smooth, no shape elements		No hole, no breakthrough	No surface machining	No gear teeth	No auxiliary hole
1		$0.5 < L/D < 3$	Stepped to one end		No shape elements	Smooth or stepped to one end	No shape elements	Surface plane and/or curved in one direction, external		Axial, not on pitch circle diameter
2		$L/D \geq 3$		or smooth	Thread		Thread	External plane surface related by graduation around a circle		Axial on pitch circle diameter
3					Functional groove		Functional groove	External groove and/or slot		Radial, not on pitch circle diameter
4			Stepped to both ends		No shape elements	Stepped to both ends	No shape elements	External spline (polygon)		Axial and/or radial and/or other direction
5					Thread		Thread	External plane surface and/or slot, external spline		Axial and/or radial on PCD and/or other directions
6	Nonrotational parts				Functional groove		Functional groove	Internal plane surface and/or slot	With gear teeth	Spur gear teeth
7					Functional cone		Functional cone	Internal spline (polygon)		Bevel gear teeth
8					Operating thread		Operating thread	Internal and external polygon, groove and/or slot		Other gear teeth
9					All others		All others	All others		All others

TABLE 6 (Continued).

Digit 1 — Component class

		Component class
	Rotational components	
	Rotational components	
	Rotational components	
3	Rotational components	$L/D \leq 2$ with deviation
4	Rotational components	$L/D > 2$ with deviation

Digit 2 — Overall shape

		Overall shape
0	Around one axis, no segments	Hexagonal bar
1	Around one axis, no segments	Square or other regular Polygonal section
2	Around one axis, no segments	Symmetrical cross section producing no unbalance
3	Around one axis, no segments	Cross sections other than 0 to 2
4		Segments after rotational machining
5		Segments before rotational machining
6	Around more than one axis	Rotational components with curved axis
7	Around more than one axis	Rotational components with two or more parallel axes
8	Around more than one axis	Rotational components with intersecting axes
9		Others

Digit 3 — Rotational machining

		Rotational machining
0		No rotational machining
1	External shape	Machined
2	External shape	With screw thread(s)
3	Internal shape	Smooth
4	Internal shape	Stepped toward one or both ends (multiple increases)
5	Internal shape	With screw threads
6	External and internal shape	Machined
7	External and internal shape	Screw thread(s)
8		External shape elements
9		Other shape elements

Digit 4 — Plane surface machining

Code	Plane surface machining
0	No surface machining
1	External plane surface and/or surface curved in one direction
2	External plane surfaces related to one another by graduation around a circle
3	External groove and/or slot
4	External spline and/or Polygon
5	External plane surface and/or slot and/or groove, spline
6	Internal plane surface and/or groove
7	Internal spline and/or Polygon
8	External and internal spline and/or slot and/or groove
9	Other

Digit 5 — Auxiliary hole(s), gear teeth, forming

			Auxiliary hole(s), gear teeth, forming
0			No auxiliary holes, gear teeth and forming
1	No forming, no gear teeth		Axial hole(s) not related by drilling pattern
2	No forming, no gear teeth		Holes axial and/or radial and/or in other directions, not related
3	No forming, no gear teeth	Related by a drilling pattern	Axial holes
4	No forming, no gear teeth	Related by a drilling pattern	Holes axial and/or radial and/or in other directions
5	Forming, no gear teeth		Formed, no auxiliary holes
6	Forming, no gear teeth		Formed, with auxiliary holes
7			Gear teeth, no auxiliary holes
8			Gear teeth, with auxiliary hole(s)
9			Other

TABLE 6 (Continued).

Digit 1			Digit 2		Digit 3		Digit 4	
Diameter D or edge length A			Material		Initial form		Diameter D or edge length A	
	mm	inches						
0	≤ 20	≤ 0.8	0	Cast iron	0	Round bar, black	0	No accuracy specified
1	> 20 ≤ 50	> 0.8 ≤ 2.0	1	Modular graphitic cast iron and malleable cast iron	1	Round bar, bright drawn	1	2
2	> 50 ≤ 100	> 2.0 ≤ 4.0	2	Mild steel ≤ 26.5 tonf/in^2 not heat treated	2	Bar: triangular, square, hexagonal, others	2	3
3	> 100 ≤ 160	> 4.0 ≤ 6.5	3	Hard steel > 26.5 tonf/in^2 heat-treatable low-carbon and case-hardening steel, not heat treated	3	Tubing	3	4
4	> 160 ≤ 250	> 6.5 ≤ 10.0	4	Steels 2 and 3 heat treated	4	Angle, U–, T–, and similar sections	4	5
5	> 250 ≤ 400	> 10.0 ≤ 16.0	5	Alloy steel (not heat treated)	5	Sheet	5	2 and 3
6	> 400 ≤ 600	> 16.0 ≤ 25.0	6	Alloy steel heat treated	6	Plate and slabs	6	2 and 4
7	> 600 ≤ 1000	> 25.0 ≤ 40.0	7	Nonferrous metal	7	Cast or forged components	7	2 and 5
8	> 1000 ≤ 2000	> 40.0 ≤ 80.0	8	Light alloy	8	Welded assembly	8	3 and 4
9	> 2000	> 80.0	9	Other materials	9	Premachined components	9	2 + 3 + 4 + 5

* Length/Diameter ratio is used to classify rotational components

* Length/Width and Length/Height ratios are used to classify non-rotational components.

The Opitz geometric code uses 5-digits, representing:

a. Component Class

b. Basic Shape

c. Rotational-Surface Machining

d. Plane-Surface Machining

e. Auxiliary Holes, Gear Teeth, and Forming

Primary, secondary, and auxiliary shapes can be represented using the five geometric digits.

A supplemental code containing four digits is usually appended to the Opitz code. The first digit represents the major dimension (either diameter or edge length). The approximate component size can then be determined by using the dimension ratio specified in the geometry. The dimension range is specified from 0.8 to 80 in. Dimensions of less than 0.8 in. and greater than 80 in. are represented by a 0 or 9 code, respectively. The material type, raw material shape, and accuracy are represented by digits 2,3, and 4. The Opitz code is concise and easy to use. It has been adopted by many companies as their coding subsystem. Several CAM-1 CAPP systems currently use an Opitz-based coding system.

6.5.2. The Vuoso-Praha System

The Vuoso-Praha code is a 4-digit coding system that characterizes a part by Kind, Class, Group, and Material. The code is illustrated in Table 7.

This type of coding is typically used for rough part classification so as to identify the type of department that would produce the part. For example, the part shown in Figure 33 would be classified as

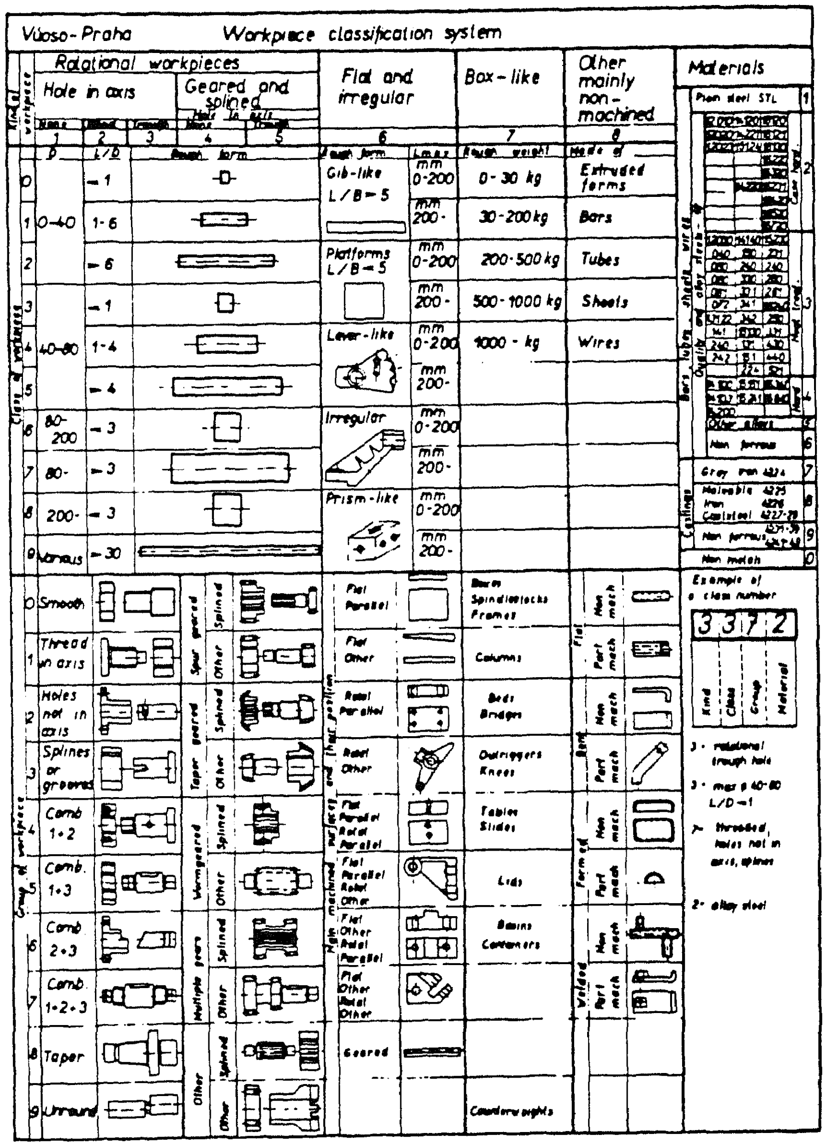
3 Rotational workpiece, with through hole

3 D = 51.2 mm; L/D = 0.6

0 Smooth

? Material not specified

TABLE 7. THE VUOSO-PRAHA CODE



6.5.3. The KK-3 System

The KK-3 coding system is a general-purpose classification and coding system for machining parts. KK-3 was developed by the Japan Society for the Promotion of Machine Industry (JSPMI) (Japan Society, 1980). Parts to be classified are primarily metal-cutting and grinding components. KK-3 was first presented in 1976, and uses a 21-digit decimal system. The code structure for rotational components is shown in Tables 8 and 9.

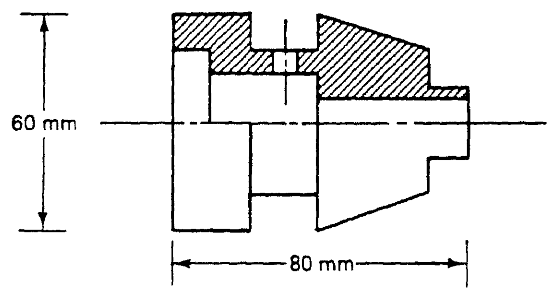
TABLE 8. STRUCTURE OF THE KK-3 CODING SYSTEM (ROTATIONAL COMPONENTS)

Digit	Items	(Rotational components)	
1	Parts name	General classification	
2		Detail classification	
3	Materials	General classification	
4		Detail classification	
5	Chief dimensions	Length	
6		Diameter	
7	Primary shapes and ratio of major dimensions		
8	Shape details and kinds of processes	External surface	External surface and outer primary shape
9			Concentric screw threaded parts
10			Functional cut-off parts
11			Extraordinary shaped parts
12			Forming
13			Cylindrical surface
14		Internal surface	Internal primary shape
15			Internal curved surface
16			Internal flat surface and cylindrical surface
17		End surface	
18		Nonconcentric holes	Regularly located holes
19			Special holes
20		Noncutting process	
21	Accuracy		

TABLE 9. FUNCTIONAL DELINEATION OF THE KK-3 CODING SYSTEM

I \ II			0	1	2	3	4	5	6	7	8	9	II / I
0	Rotational components	Gears	Spur, helical gear(s)	Internal gear(s)	Bevel gear(s)	Hypoid gear(s)	Worm gear(s)	Screw gear(s)	Sprocket wheel	Special gear	Round vessel	Other(s)	Gears
1	Rotational components	Shafts, spindles	Spindle, arbor, main drive	Counter shaft	Lead screw(s)	Screwed shaft	Round rod(s)	Eccentric shaft(s)	Splined shaft	Cross shaft	Round column	Round casting	Shafts, spindles
2	Rotational components	Main drive	Pulley(s)	Clutch	Brake(s)	Impeller(s)	Piston(s)	Round tables	Other(s)	Flange	Chuck(s)	Labyrinth seal(s)	Main drive
3	Rotational components	Guiding parts	Sleeves, bushing	Bearing metal	Bearing(s)	Roller(s)	Cylinder	Other(s)	Dial plate(s)	Index plate(s)	Cam(s)	Others	Guiding parts
4	Rotational components	Fixing part	Collar(s)	Socket, spacer	Pin(s)	Fastening screws	Other(s)	Handles	Spool(s)	Round links	Screw(s)	Others	Fixing part
5	Nonrotational												
6	Nonrotational												
...	Nonrotational												
9	Nonrotational												

Because KK-3 is much greater in length than Opitz and CODE, more information can be represented.



Code digit	Item	Component condition	Code
1	Name	Control valve	0
2		(others)	9
3	Material	Copper bar	7
4			
5	Dimension length	80 mm	2
6	Dimension diameter	60 mm	2
7	Primary shape and ratio of chief dimension	L/D 1.3	2
8	External surface	With functional tapered surface	3
9	Concentric screw	None	0
10	Functional cutoff	None	0
11	Extraordinary shaped	None	0
12	Forming	None	0
13	Cylindrical surface ≥ 3	None	0
14	Internal primary	Piercing hole with dia. variation, NO cutoff	2
15	Internal curved surface	None	0
16	Internal flat surface	None	0
17	End surface	Flat	0
18	Regularly located hole	Holes located on circumferential line	3
19	Special hole	None	0
20	Noncutting process	None	0
21	Accuracy	Grinding process on external surface	4

FIGURE 36. Example Of A KK-3 Coding System.

The KK-3 code includes two digits for the component-name (functional-name) classification. The first digit classifies the general function, such as gears, shafts, drive, and moving parts, and fixing parts. The second digit describes more detailed functions. For example, included in a single family, there are spur gears, bevel gears, worm gears, and so on. With two digits, KK-3 can classify 100 functional names for rotational and nonrotational components. However, at times, this can be as confusing as it is complete. KK-3 also classifies materials using two-code digits. The first digit classifies material type and the second digit classifies shape of the raw material. Dimensions and dimension ratios are also classified. Some redundancy can be found, that is, length, diameter, and length/diameter ratios are classified in KK-3 using 13 digits of code (much more detail than either the Opitz or CODE systems). An example of coding a component using KK-3 is illustrated in Figure 36.

6.5.4. The CODE System

The Code system is a parts classification and coding system developed and marketed by Manufacturing Data Systems, Inc. (MDSI), of Ann Arbor, Michigan. Its most universal application is in design engineering for retrieval of part design data, but it also has applications in manufacturing process planning, purchasing, tool design, inventory control.

The CODE system is a Hexadecimal (8-digits) coding scheme. For each digit there are 16 possible values (0,1,....,9,A,....,F) which are used to describe the part's design and manufacturing characteristics. The initial digit position indicates the basic geometry of the part and is called the Major Division of the CODE system. This digit would be used to specify whether the shape was a cylinder, flat piece, block, or other. The interpretation of the remaining seven digits depends on the value of the first digit, but these remaining digits form a chain-type structure. Hence the CODE system possesses a hybrid structure.

The second and third digits provide additional information concerning the basic geometry and principal manufacturing process for the part. Digits 4,5, and 6 specify secondary manufacturing processes such as threads, grooves, slots, and so forth. Digits 7 and 8 are used to indicate the overall size of the part (e.g., diameter and length of the turned part) by classifying it into one of 16 size ranges for each of two dimensions. Figure 37 shows a portion of the definitions for digits 2 through 8, given that the part has already been classified as a cylindrical geometry (Major Division 1 for concentric parts other than profiled).

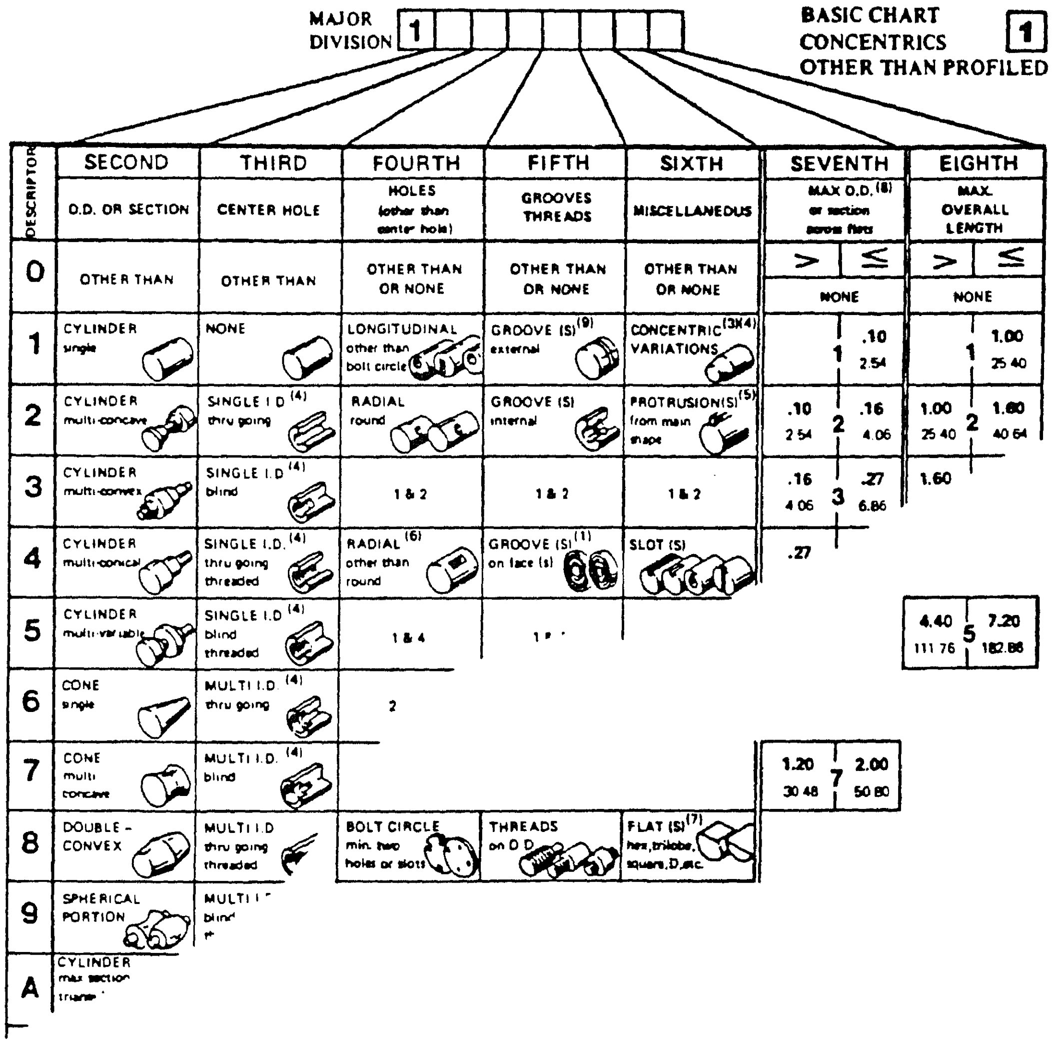
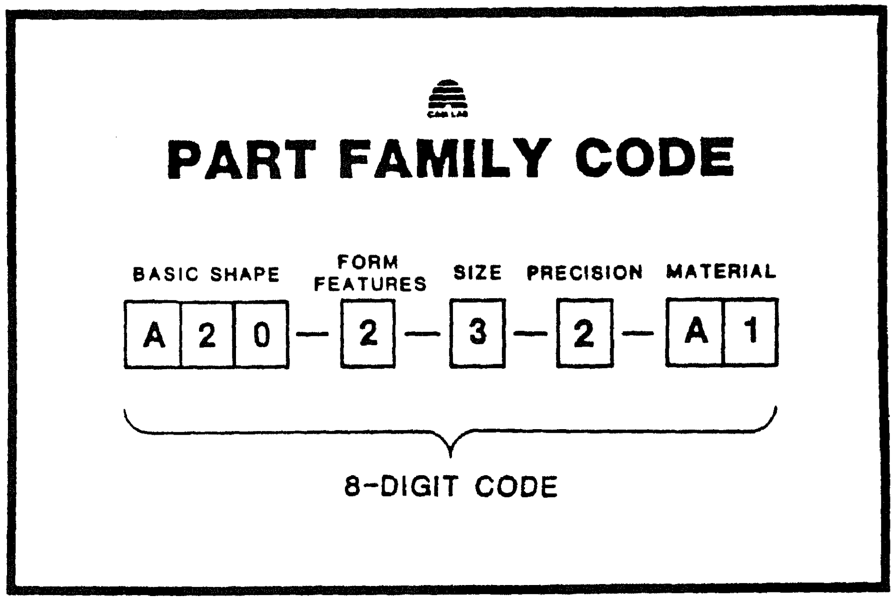


FIGURE 37. A Portion Of The CODE System Of MDSI.

6.5.5. DCLASS Systems



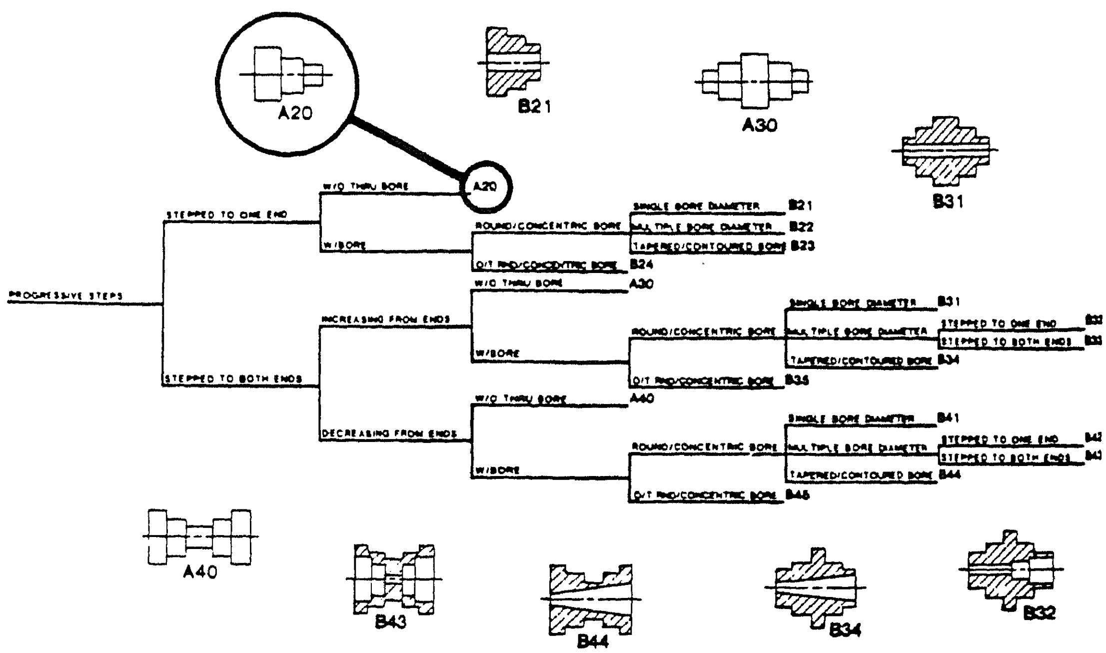


FIGURE 38. DCLASS Structure.

The DCLASS system was developed by Del Allen at Brigham Young University (Allen and Smith, 1980). DCLASS was intended to be decision-making and classification system (thus the name DCLASS). It is a tree-structured system, as it can be seen in Figure 38, it can generate codes for components, materials, processes, machines, and tools. For components, an 8-digit code is used:

Digits	**1-3**	**Basic Shape**
Digit	**4**	**Form Feature**
Digit	**5**	**Size**
Digit	**6**	**Precision**
Digits	**7 & 8**	**Material**

In DCLASS, each branch represents a condition, and a code can be found at the terminal (junction) of each branch. Multiple passes of the decision tree allow a complete code to be found. Figure 38 illustrates the process by showing a portion of the decision tree for components. One pass on this tree generates the first three digits of the code. Code construction is established using certain roots. (N nodes and E nodes). At N nodes, all branches can be true. At E nodes, only one branch can be true.

6.5.6. The MICLASS System

MICLASS stands for Metal Institute Classification System and was developed by TNO, the Netherlands Organization for Applied Scientific Research. It was started in Europe about five years before introduced in United States in 1974. Today, it is marketed in the United States by the Organization for Industrial Research in Waltham, Massachusetts. The MICLASS system was developed to help automate and standardize a number of design, production, and management functions. These include:

* Standardization of engineering drawings
* Retrieval of drawings according to classification number
* Standardization of process routing
* Automated process planning
* Selection of parts for processing on particular groups of machine tools
* Machine tool investment analysis

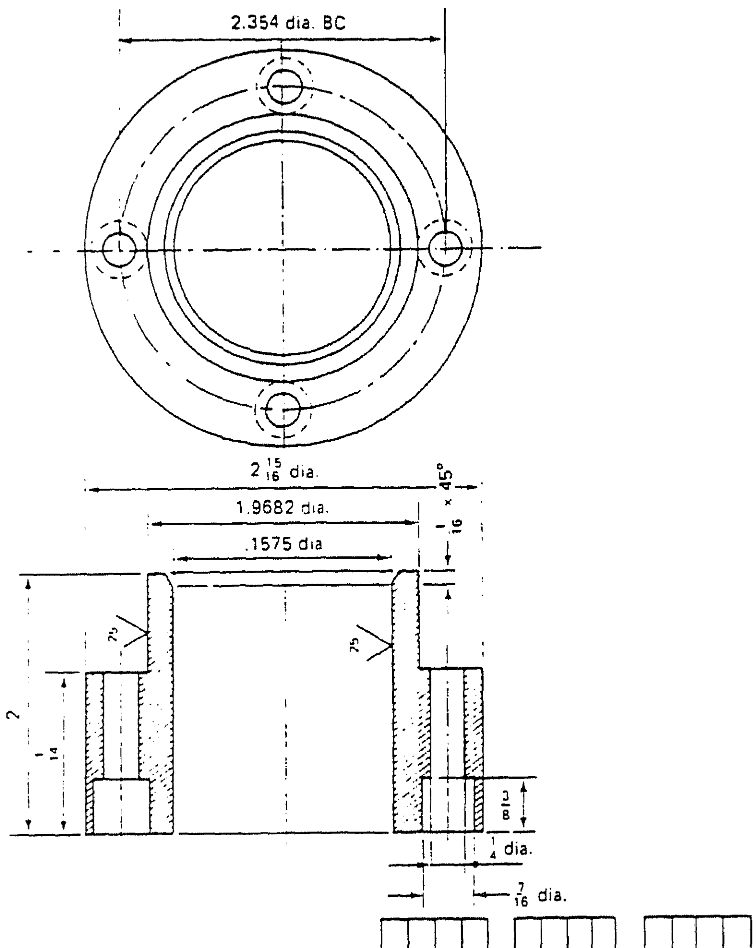
The MICLASS classification number can range from 12 to 30 digits. The first 12 digits are a universal code that can be applied to any part. Up to 18 additional digits can be used to code data that are specified to the particular company or industry. For example, lot size, piece time, cost data, and operation sequence might be included in the 18 supplementary digits.

The workpart attributes coded in the first 12 digits of the MICLASS number are as follows:

Digit	**1**	**Main shape**
Digits	**2 & 3**	**Shape elements**
Digit	**4**	**Position of shape element**
Digits	**5 & 6**	**Main dimensions**
Digit	**7**	**Dimension ratio**
Digit	**8**	**Auxiliary dimension**
Digits	**9 & 10**	**Tolerance code**
Digits	**11 & 12**	**Material codes**

One of the unique features of the MICLASS system is that parts can be coded using a computer interactively. To classify a given part design, the user responds to a series of questions asked by the computer. The number of questions depends on the complexity of the part. For a simple part, as few as seven questions are needed to classify the part. For an average part, the number of questions ranges between 10 and 20. On the basis of the responses to its questions, the computer assigns a code number to the part. Because the system developer, TNO, is an international organization, the program was written to converse in any of four languages:

English —— French —— German —— Dutch



DRAWING	TOLERANCES	MATERIAL
TITLE		CC15
BUSHING	Fractional ± 1/64	
DRAWING NO.	Decimal ± 0.003	125 (25) ALL OVER EXCEPT AS NOTED
7		

FIGURE 39. Workpart.

VERSION –A–

3 MAIN DIMENSIONS (WHEN ROT. PART D.L AND 0) 2.9375 2 0
DEVIATION OF ROTATIONAL FORM? NO
CONCENTRIC SPIRAL GROOVES? NO

TURNING ON OUTERCONTOUR (EXCEPT ENDFACES)? YES
SPECIAL GROOVES OR CONE(S) IN OUTERCONTOUR? NO
ALL MACH. DIAM. AND FACES VISIBLE FROM ONE END (EXC. ENDFACE - GROOVES

TYPING ERROR, ANSWER AGAIN? YES

INTERNAL TURNING? YES
INTERNAL SPECIAL GROOVES OR CONE(S)? NO
ALL INT. DIAM. + FACES VISIBLE FROM 1 END (EXC. GROOVES)? YES

ALL DIAM. + FACES (EXC. ENDFACE) VISIBLE FROM ONE SIDE? YES

ECC. HOLING AND/OR FACING AND/OR SLOTTING? YES
IN INNERFORM AND/OR FACES (INC. ENDFACES)? YES
IN OUTERFORM? NO

ONLY KEYWAYING ETC.? NO

MACHINED ONLY ONE SENSE? YES
ONLY HOLES ON A BOLTCIRCLE AT LEAST 3 HOLES? YES

FORM-OR THREADING TOLERANCE? NO

DIAM. ROUGHNESS LESS THAN 33 RU (MICRO-INCHES)? YES
SMALLEST POSITIONING TOL. FIELD?. .016
SMALLEST LENGTH TOL. FIELD? .0313

MATERIAL NAME? CC15

CLASS. NR. = 1271 3231 3144

DRAWING NUMBER MAX 10 CHAR? 7

NOMENCLATURE MAX 15 CHAR? BUSHING

CONTINUE (1), STOP (2), SECOND PART AGAIN (3)? 2

PROGRAM STOP AT 4690

USED ________ UNITS

FIGURE 39 (Continued). Computerized MICLASS System.

Determination Of Code Number For Workpart.

Also it can operate in either inches or metric, or both. MICLASS is also one of the earliest interactive coding systems, Figure 39 illustrates an interactive coding session. MICLASS has been adapted by many U.S. industries. Several application programs based on MICLASS are currently available, such as the MULTIPLAN and MULTICAPP variant process-planning systems.

6.6. Benefits Of Group Technology

Broadly defined, Group Technology (GT) is the application of knowledge about groups. More specially, GT implies the notion of recognizing and exploring and exploiting similarities in three distinct way:

a. By performing like activities together

b. By standardizing similar tasks

c. By efficiently storing and retrieving information about recurring problems.

It should be regarded not as a tool or an application but as a management philosophy. In particular, GT is not a classification and coding system, nor is it equivalent to cellular manufacturing; the first is a tool to help implement GT, the second is an application of GT.

Although group technology is expected to be an important principle in future production plants, it has not yet achieved the widespread application which might be expected. While there is currently a high level of interesting GT, fairly little is known about its uses. There are several reasons for this. First, there is a problem of rearranging the machines in the plant into GT cells. Many companies have been inhibited from adopting GT because of the expense and disruption associated with the transition to GT machines cells. Second, there is the problem of identifying part families among the many components produced in the plant. Usually associated with this problem is the expense of parts classification and coding. Not only this procedure is expensive, but it also requires a considerable investment in time and personnel resource. Managers often feel that these limited resources can be better be allocated to other projects than group technology with its uncertain future benefits. Finally, it is common for companies to encounter a general resistance among its operating personnel when changeover to a new system is contemplated.

When these problems are solved and group technology is applied, the company will typically realize benefits in the following areas:

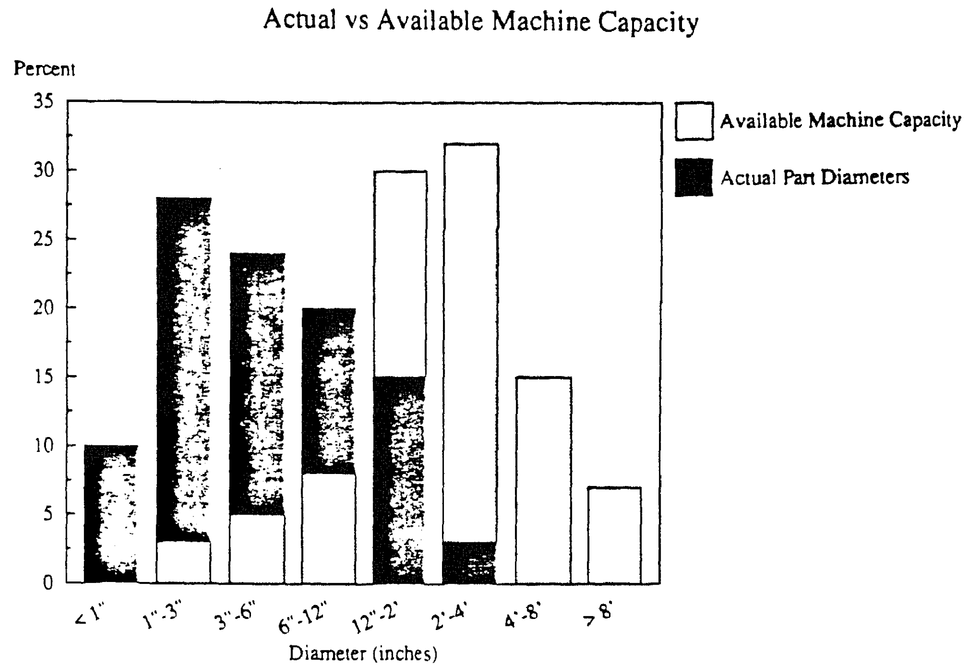
* Product design
* Tooling and setups
* Materials handling
* Production and inventory control
* Employee satisfaction
* Process planing procedures

6.7. Machine Selection

A common characteristic of U.S. industry is underutilization of expensive processing equipment. The underutilization can take two forms:

a. Much of the machine time is idle and totally unproductive;

b. Many of the parts assigned to a specific machine are far below the capacity of the machine.



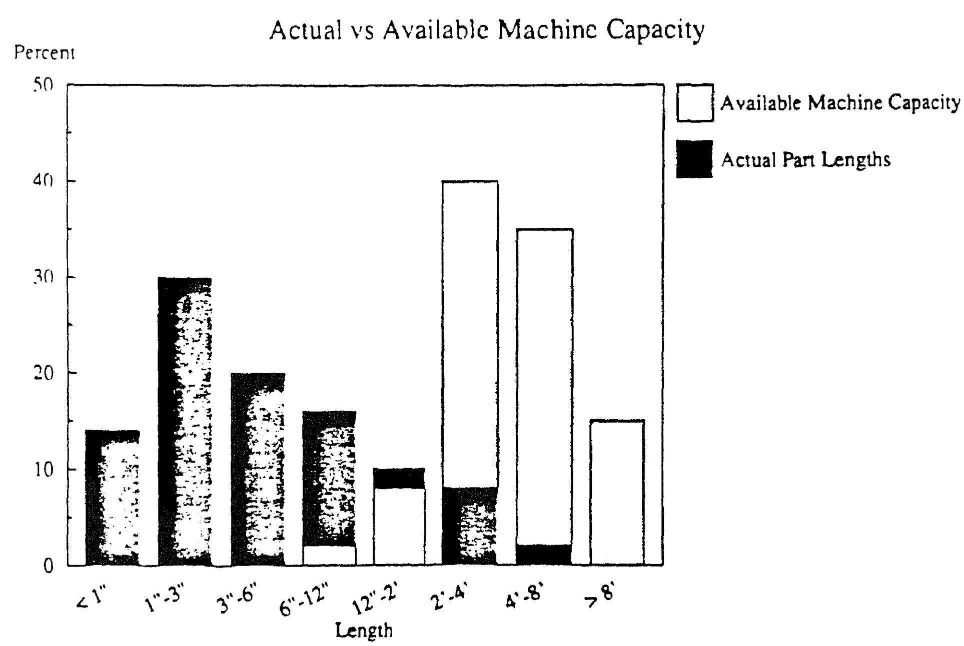


FIGURE 40. Comparison (Typical) Of A Turned-Part Dimension As A Function Of Machine Capacity.

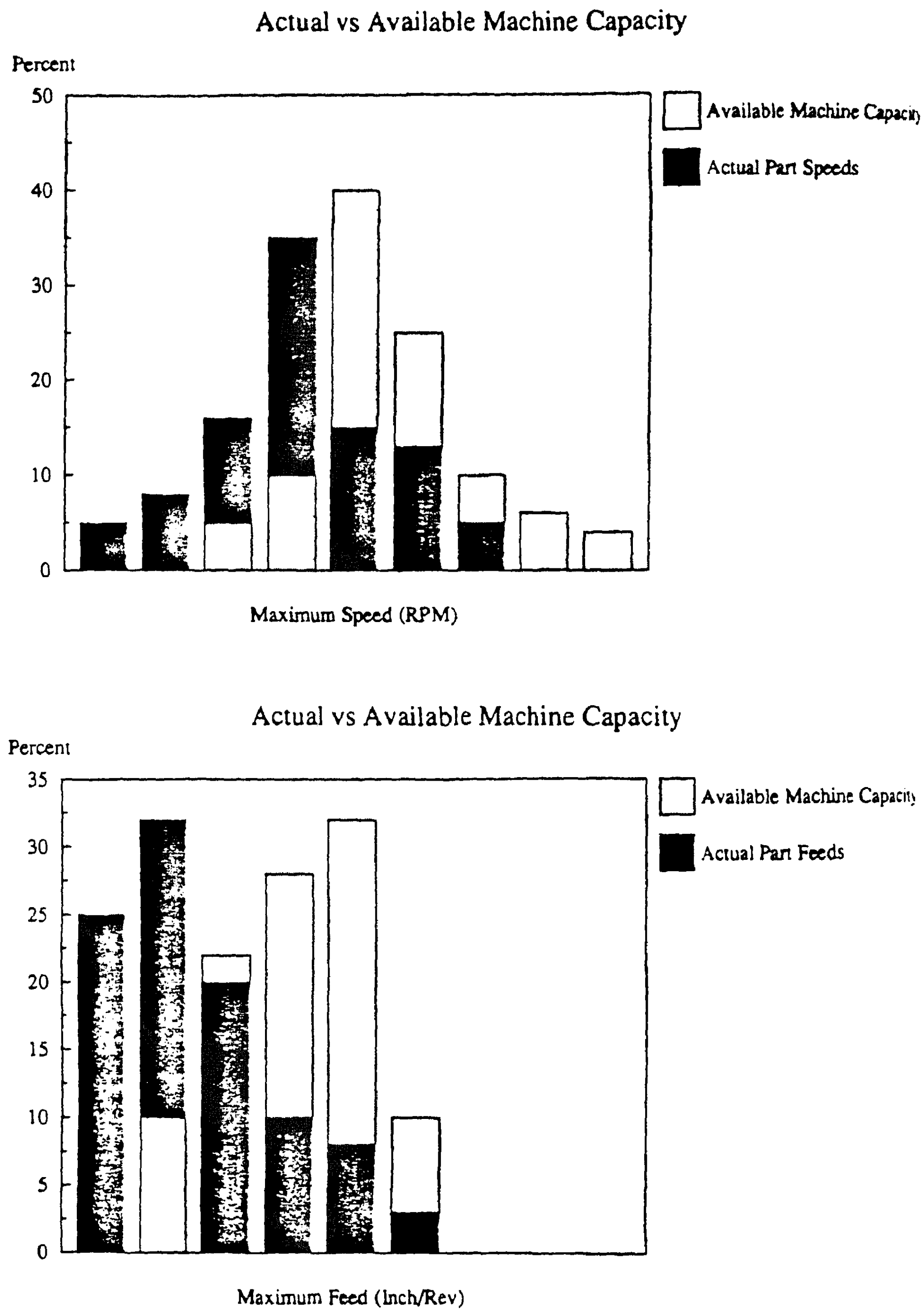


FIGURE 41. Comparison Of Maximum Speeds And Feeds (Machine Capacity) With Maximum Used.

By grouping closely matched parts into a part family, machines can be more fully utilized from both a scheduling as well as a capacity standpoint.

By using a part coding system, similar parts with similar feature dimension specifications can be assigned to the same part family , and machines corresponding to the minimum product specification can be selected rather than overspecifying (and overpaying) the processing. Figure 40 illustrates this phenomenon for lathe parts. In the figure, it can be seen that only few percent of the parts being machined require the full lathe swing or length. Furthermore, the speed and feed capacity of the lathe can also be over-specified. See Figure 41. Since additional features on machine tools require additional cost, be careful not to over-specify a machine and not to use the features.

CHAPTER 7

7. CAPP Using Artificial Intelligence

Artificial intelligence (AI), is a new and rapidly developing field of computer science. It deals with computers performing human like functions such as reasoning and interpretation. AI may also take the form of simulating the human senses of vision, hearing, touch, and even smell. These senses usually serve as a source of information for the human brain to use in making observations, drawing conclusions, problem solving, natural language processing, and formulating decisions. Computer systems are being developed which operate with the same basic approach to problem solving, rather than the fixed, sequential, programmed logic of conventional computers.

Recent advances in the field of AI demonstrate that practical AI based systems are possible. AI has achieved considerable success in the development of expert systems since the mid-1960s. Edward Feigenbaum of Stanford University, one of the leading researchers in expert systems, has defined an expert system as:

"an intelligent computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solution. Knowledge necessary to perform at such a level, plus the inference procedures used, can be thought of as a model of the expertise of the best practitioners of the field".

This area of AI has concentrated on the construction of high-performance programs in specialized professional domains--a pursuit that has encouraged an emphasis on the knowledge that underlies human expertise and has simultaneously decreased the apparent significance of domain-independent problem-solving theory. A new set of principles, tools, and techniques has emerged that forms the basis of knowledge engineering, which is a new term adopted by the researchers in this field in order to describe this new discipline.

7.1. AI and Process Planning

Today, with rapidly diminishing number of experienced process planners in industry, there is an urgent need to automate the process-planning function. Many computer-aided process planning (CAPP) systems have been developed as a result of this urgency. The complexity, in addition to the variety of tasks in preparing a process plan, requires a significant amount of time from an experienced process planner in almost all existing CAPP systems.

The successful application of artificial intelligence in many science and engineering areas reveals that AI is applicable to the process planning and benefits have been reported.

A productive CAPP system must contain a tremendous amount of knowledge--facts about the machine shop and rules about arranging machine operations. Furthermore, the system should be flexible because facts and rules in the database require constant updating. This is especially true in today's manufacturing environment. In a traditional CAPP system, manufacturing knowledge is coded line by line in the program's statements. Any modification to the facts and rules require rewriting the original program. In other words, a traditional CAPP program cannot learn new knowledge unless it is explicitly rewritten. The inflexibility of traditional methodoligies has retarded the implementation of CAPP systems, which are the most important factors in CAD/CAM linkages.

An expert system stores knowledge in a special manner so that it is possible to allow pieces of knowledge to be added, delete, and modify from the system without having to modify other segments of the system. Such a system would allow initial knowledge to be encoded into the system, and once the system was operating, it could be augmented by additional information. A list of operations could be created for the surface to be machined. The prerequisites and restrictions of each process must be considered before it is performed. A software control structure must search through the knowledge base until a satisfactory process is identified. The system must interact with the user until it has determined the process to produce every surface.

Wolfe and Kung describe the development of a generative process plan using an expert system methodology. The system aims to extract form features of a part from a CAD System. These features, along with details about tolerance and surface finish, are placed in a data base which is used by expert system to generate a process plan. Perhaps the earliest plan generator to be developed using AI techniques was GARI. GARI showed that as the number of production rules increased, the possibility of conflicts between them also increased. This would require appropriate conflict breaking methods to be enunciated. In addition, it is possible that a rule may also have an exception which may lead to delays due to tedious conflict resolution.

Hummel developed an expert system to generate process plans to machine a part in one fixturing setup at a machine tool. The use of localized process planning capabilities for each control module in a distributed computing atmosphere is recommended. This would require a hierarchical network of independent modules whose planning spectrum would reflect the span of the control at that level in the hierarchy. Each module would have a local knowledge base and a set of requirements that would be machined with the capabilities of an individual machining tool. It is imperative that each module in the hierarchy be able to access the knowledge bases of other modules in the system. The machine tool planner developed could work as a segment of a larger system, responsible for generating machining instructions required to process a part in a single fixturing setup at a machine tool, or as a stand alone system, dedicated to a single machine.

CHAPTER 8

8. Process Planning Systems - A Survey

There exist today both variant and generative process plans. Process planning systems have been historically been developed by a single organization or in an academic atmosphere. Following is the list of some of the CAPP systems that have been described in the open literature. The list is not intended to be exhaustive, nor are the attributes intended to be critical judgement.

8.1. APPAS

Generative system, exclusively for parts produced on machining centers. Uses a special code to provide technical data and information describing each surface. Complex decision tree structure used.

8.2. AUTOPLAN

Generative system, provides additional details such as jig and fixture selection, and material and stock determination. Does not claim to be CAPP system, instead a system to aid process planner. Uses graphical capability.

8.3. CAM-I CAPP

A database management system with appropriate retrieval logic. Interactive with user specified GT coding technique.

8.4. ACAPS

A semi-generative system that allows interactive coding of surfaces. Considers available capacity and suggests a possible machining sequence. Can handle holes, slots, plane and turned surfaces.

8.5. ICAPP

Developed for milling, drilling, and boring, the sequence of operations is decided by the part's features, material, dimensions and tolerances. Calculates speeds, operation and production times.

8.6. AUTOCAPP

An interactive system for process planning of turned parts, output includes speeds, feeds, number of cuts, operations, handling and machining time. Planner has to indicate surfaces to be machined, accuracy and surface finish.

8.7. MIPLAN

A special purpose, interactive word processor that allows a planner to construct and edit process plans from standard plans. New plans can be created, stored plans retrieved and edited, or an incomplete plan completed.

8.8. GARI

An expert system, it consists of a knowledge base and a general purpose problem solver. Knowledge stored as production rules. Successive iterations used to generate plan.

8.9. GENPLAN

A Generative system, capable of instantly creating work instructions for the manufacturing of aircraft parts and subassemblies. Analyzes a part geometry, material, and specifies tools and processes needed.

8.10. SHAPES

Part profiles are entered in 'SHAPE' elements. The data on the material and component size are used to determine the machine tool, and collets, chucks, and centers to hold workpiece.

8.11. TIPPS

Generative system, that addresses model decomposition and shape identification. The facility's capability and required processes are matched to devise a process plan. A combination of AI and decision tree approaches are used. Inputs are in the form of CAD boundary representation.

8.12. TOJICAPP

System for rotational parts. Combines generative and variant planning. Can be used to generate a new plan interactively. Outputs include cutting parameters, time estimates, production times, and cost estimates.

8.13. MICRO-CAPP

Interactive variant system, system commands are all two letter neumonics. Coding and part classification is through GT.

8.14. GEOMETRIC PROGRAMMING

Techniques used to select optimal machining parameters for a wide variety of operations. Inputs required-speed, machine tool, feed, and HP. Can optimize cost or time. Does not determine operation sequence or the machines to be used.

8.15. CAPP and Dynamic Process Selection Techniques

The primary objective of most process planning systems has been to develop a sequence of operations based upon technical and economical criteria. This does not consider the dynamic conditions of the machine shop. A versatile, flexible manufacturing system would require a CAPP that was dynamic in nature. The system must be able to generate alternative, feasible plans. Dynamic process selection would involve the capacity to decide which process would be used to manufacture a part, taking into consideration current facility status. This assumes the alternative processes exist to machine a part. This means that a part that could be processed on a particular class of machines could also be processed on a different class of machines using significantly different manufacturing procedures.

Nof and Barash state that production rate and machine utilization are two criteria to be monitored in a facility. It was found that when dynamic allocation of either the original plan or the alternative were undertaken, the production rate and the machine utilization improved. However, product quality may be dependent upon the manufacturing process used. In a particular facility, since most parts are not manufactured just once, but repetitively, it is a sound decision to prepare alternate process plans for each part.

CHAPTER 9

9. Description of SORICH CAPP System

The SORICH CAPP System is not a specialized software package as the other CAPP systems available on the market. It is a customized macro based program using the spreadsheet software package Quattro Pro. It is a concept that is being demonstrated based on the ideas of CAPP. This type of CAPP system can be designed on any of the spreadsheet softwares available on the market, such as but not limited to; Quattro Pro, Lotus 1-2-3, V-P Planner and Excel. The system must be designed around a particular company's manufacturing ideas and philosophies.

9.1. Design of System

The SORICH CAPP system was designed for use with the spreadsheet package Quattro Pro, but as indicated above it can be adapted to any spreadsheet package. The system is designed and written in the same way as any program would be done, using the MACROS, Table 10, and by following logical programming practices and the limitations and coding set out by the software being used. SORICH was designed to be as user friendly as possible. The need was for the user to be able to enter the desired information with limited room for error. Several different attempts were made at reaching this goal. There were no real step by step instructions to follow and trial and error and continued searching through the Quattro Pro manual was needed to complete this task.

Upon running the program, the first step to occur is the removal of any existing information from a previous run, this is performed automatically by the macro. The macro then accesses custom menus that prompt the user to make selections. The custom menus are designed by the programer, when accessed they resemble a typical pull down menu similar to those in most commercial softwares. The user can make his selection from the menus and these selections then determine the code number of the part to be produced. The macro will then allow the user to import, view, alter, save and print the Route Sheets and Bill of Materials for the particular code number selected.

TABLE 10. MACROS OF SORICH CAPP SYSTEM

```
MACRO {;THE FOLLOWING 4 COMMANDS}                              *
CAPP  {;DELETE THE OLD INFORMATION}                            *
      {;TO ALLOW FOR NEW INFO}                                 *
      {BLANK CHOICES}                                          *
      {BLANK ROUTE}                                            *
      {BLANK BOM}                                              *
      {BLANK DRAWING}                                          *
      {}                                                       *
      {;THE FOLLOWING DISPLAYS A}                              *
      {;MESSAGE UNTIL ENTER IS HIT}                            *
      {}                                                       *
      {MESSAGE MSG1,15,10,0}                                   *
      {}                                                       *
      {;THE FOLLOWING 7 COMMANDS}                              *
      {;BRING UP CUSTOM MENUS YOUR}                            *
      {;CHOICES DETERMINE THE CODE}                            *
      {;NUMBER FOR THE NEW PART}                               *
      {}                                                       *
      {MENUCALL TYPE MENU}                                     *
      {IF CHOICE1<>3}{BRANCH CAPP}                             *
      {MENUCALL SHAPE MENU}                                    *
      {MENUCALL SIZE MENU}                                     *
      {MENUCALL FINISH MENU}                                   *
      {IF CHOICE4=1}{MENUCALL WOOD MENU}                       *
      {IF CHOICE4=2}{MENUCALL COLOR MENU}                      *
      {IF CHOICE4=3}{MENUCALL METAL MENU}                      *
      {MENUCALL TOP MENU}                                      *
      {MENUCALL CASTER MENU}                                   *
      {}                                                       *
      {;THE FOLLOWING 3 LINES }                                *
      {;CONVERT USERS RESPONSES}                               *
      {;INTO A VALUE}                                          *
      {ESC}                                                    *
      /EVCODE`CODE1~                                           *
      /EVCODE`CODE2~                                           *
      /EVCODE~CODE3~                                           *
      {}                                                       *
      {;THE FOLLOWING 3 COMMANDS}                              *
      {;BRING UP CUSTOM MENUS}                                 *
      {}                                                       *
      {MENUCALL IMPORT MENU}                                   *
      {MESSAGE MSG2,15,10,0}                                   *
      {MENUCALL VIEW MENU}                                     *
      {MENUCALL SAVE MENU}                                     *
      {MENUCALL PRINT MENU}                                    *
      {HOME}                                                   *
      {QUIT}                                                   *
```

TABLE 10. (Continued)

```
MSG1  DURING EXECUTION OF CUSTOM              *
      MENU'S YOU MUST HIGHLIGHT               *
      YOUR CHOICE BY USE OF THE               *
      ARROW KEYS AND HIT ENTER.               *
      ** HIT ENTER TO CONTINUE **             *
                                              *
MSG2  UPON CHOOSING TO VIEW YOU WILL HAVE     *
      FREEDOM TO MOVE THROUGHOUT THE          *
      SPREADSHEET WITH THE ARROW KEYS AND     *
      AND MAKE ANY NECESSARY CHANGES.         *
      TYPE IN CHANGE AND HIT ARROW KEY        *
      HIT ENTER WHEN ALL CHANGES ARE MADE.    *
      ** HIT ENTER TO CONTINUE **             *
                                              *
MSG4  INFORMATION FOR THIS SELECTION IS       *
      UNDER PROCESS, MAKE ANOTHER SELECTION.  *
      ** HIT ENTER TO CONTINUE **             *
                                              *
MSG5  COPY DOWN THE CODE NUMBER THAT          *
      IS DISPLAYED NEXT. USE THIS CODE        *
      NUMBER TO IMPORT FILES. IF AN           *
      EXACT MATCH DOES NOT EXIST CHOOSE       *
      THE FILE WITH THE CLOSEST NUMERIC       *
      VALUE. FILES ENDING IN "R" ARE ROUTE    *
      SHEETS AND FILES ENDING IN "B"          *
      ARE BOM'S.                              *
      ** HIT ENTER TO CONTINUE **             *
                                              *
MSG6  COPY DOWN THE CODE NUMBER THAT          *
      IS DISPLAYED NEXT. USE THIS CODE        *
      NUMBER TO SAVE THE NEW FILE CREATED     *
      END WITH "R" FOR ROUTE SHEET            *
      OR END WITH "B" FOR BOM.                *
      ** HIT ENTER TO CONTINUE **             *
                                              *
MSG7  END THE FOLLOWING CODE NUMBER           *
      WITH "R" FOR ROUTE SHEET.               *
      ** HIT ENTER TO CONTINUE **             *
                                              *
MSG8  END THE FOLLOWING CODE NUMBER           *
      WITH "B" FOR BOM.                       *
      ** HIT ENTER TO CONTINUE **             *
                                              *
MSG9  END THE FOLLOWING CODE NUMBER           *
      WITH "D" FOR DRAWING.                   *
      ** HIT ENTER TO CONTINUE **             *
```

9.2. Use Of The SORICH System

The following is a procedure for the operation of the SORICH CAPP system. SORICH is a Quattro Pro program and must be executed in a Quattro Pro environment. 5 1/4 disk that accompanies with the THESIS, has the SORICH file.

Step 1. Start Quattro Pro and load SORICH by entering the following commands:

a. / File Retrieve

b. Now type SORICH and hit enter

Step 2. Begin Macro execution by entering the following commands:

a. / Tools Macro Execute

b. Now type SORICH and hit enter<

Step 3. The customized pulldown menus will be displayed in the upper left hand corner of the screen.

Step 4. Use the arrow keys to highlight your choice.

Step 5. A brief description is given in the lower left hand corner of the screen while the choice is highlighted.

Step 6. Use the enter key to make the proper selection.

Step 7. The first seven pulldown menus are used to make selection which produces the code number.

Step 9. The following pulldown menus will allow the user to import, view, alter, save and print the route sheet and BOM

CHAPTER 10

10. CAPP - The Future

In 1984 it was forecasted that by 1990 over 50% of all process plans produced would be developed by CAPP systems. Although this seems that this ambitious estimate may not come true, the importance of CAPP systems cannot be over emphasized. The decreased in the availability of trained, experienced personnel to serve as process planners accentuates the importance of CAPP. The continuous increase in labor cost, the never ending struggle to reduce production cost, and increased productivity will force manufacturing organizations to adopt CAPP systems. Improvement in software, its transportable nature, and the advent of powerful desktop computers will be an incentive for small manufacturing organizations to adopt CAPP.

Increased efforts will be made to integrate CAD and CAM through CAPP, and reduce the time for the job to proceed from the drawing board to the production. This would cause designers and process planners to work together, designing parts with ease of manufacture as a primary consideration. It would cause the development of a unified data base that could be accessed by the process planner and the design engineer. Most CAPP systems require the user to input the specific GT code. This could be eliminated if the system could automatically create and mark surfaces. Dynamic process selection systems that suggest process plans based upon current facility load should be developed.

The recent spurt in the development and implementation of knowledge based expert systems will certainly have its effect on CAPP. The AI approach with its elegant data representation and retrieval techniques, coupled with sound decision logic programmed into system software should have widespread applications in CAPP. Applications could include processes such as metal forming, welding, casting, lesser used processes such as broaching, and traditional CAPP applications such as turning and drilling. A major hurdle would be the gathering of reliable data in order to create good, comprehensive knowledge bases.

CAPP systems have to expand their horizons to serve a greater variety of manufacturing industries. Currently, CAPP systems concentrate on machining processes, especially in processes involving turning and hole generation. Recent CAPP system applications have included operations such as cold forging, and deep drawing. Increased use of CAPP systems would make installation of CAPP systems more viable financially, and help to increase productivity levels in all manufacturing industries.

The SORICH CAPP System designed for the thesis is a product oriented software, it is a concept that is being demonstrated based on the ideas of CAPP. This type of CAPP system can be designed on any of the spreadsheet softwares available on the market. The system must be designed around a particular company's manufacturing ideas and philosophies.

CHAPTER 11

11. Conclusion

The desire to improve productivity and decrease costs have developed the need to reduce the time required by a part to go from preliminary design to actual manufacture. CAPP will reduce the time spent from product development to the commencement of production in a product's life cycle. Enhancements in available computing power coupled with reductions in computer hardware costs have made it more viable to use CAPP. Recent trends indicate that knowledge based expert systems have greatly altered CAPP's scope. It is forecasted that expert systems would be used to aid process planning involving most manufacturing processes. A hurdle to this would be the development of comprehensive data bases for the applications under considerations.

The importance of CAPP, as necessary CAD/CAM interface, is emerging. Researchers are striving to produce a generative process planning system using a CAD input interface, requiring with very little human intervention. In the future, research will employ interdisciplinary approach involving experienced process planners, design engineers, and computer scientists as well as industrial engineers.

BIBLIOGRAPHY

1. Alan, D. K., 'Generative Process Planning Using The DCLASS Information System', Monograph # 4, Computer Aided Manufacturing Laboratory, Brigham Young University, Provo, Utah, 1979.

2. Alan, D. K., & Smith, P. R., 'Computer-Aided Process Planning', Computer Aided Manufacturing Laboratory, Brigham University, Provo, Utah, 1980.

3. Amstead, B. H., Ostwald, Phillip F., & Begeman, Myron L., 'Manufacturing Processes', John Wiley & Sons, New York, 1979.

4. Berra, P. B., 'Investigation of Automated Planning And Optimization of Metal Working Processes', Doctoral Dissertation, Purdue University, West Lafayette, Indiana, 1968.

5. Caldeira, E., 'Integration of CAPP And Computerized Standard Data', Proceedings - Fall IIE Conference, 1985, PP, 106-109.

6. Chang, Tien-Chien, & Wysk, Richard A., Wang, Hsu-Pin, 'Computer-Aided Manufacturing', Prentice Hall, New Jersey, 1991.

7. Chang, Tien-Chien, 'Interfacing CAD and CAM - A Study In Hole Design ', Master Thesis, Virginia Teach, Blacksburg, Virginia, 1980.

8. Chang, Tien-Chien, Wysk, Richard A., & Davis, R. P., 'Interfacing CAD and CAM - A Study in Hole Design', Computing and Industrial Engineering, Vol. 6, # 2, 1982, PP. 91-102.

9. Chang, Tien-Chien, Wysk, Richard A., 'An Integrated CAD/Automated Process Planning System', AIIE Transactions, Vol. 13, # 3, 1981, PP. 223-233.

10. Chang, Tien-Chien, 'TIPPS-A Totally Integrated Process Planning System', Doctoral Dissertation, Virginia Teach, Blacksburg, Viginia, 1982.

11. Chang, Tien-Chien, 'Computer Aided Process Planning Today and in the Future'. Proceedings - Fall IIE Conference, Nov. 1983, PP. 349-355.

12. Chang, Tien-Chien, & Wysk, Richard A., 'Integrating CAD and CAM through Automated Process Planning', International Journal of Production Research, Vol. 22, # 5, 1984, PP. 877-894.

13. Chang, Tien-Chien, & Wysk, Richard A., 'An introduction to automated Process Planning Systems', Prentice Hall, NJ, 1985.

14. Descotte, Y., & Latombe, J. C., 'GARI: A Problem Solver that Plans How To Machine Mechanical Parts', Proceedings - IJCAI 7, Vancover, Canada, PP. 766-772, Aug. 1981.

15. El-Gomayel, J., & Abou-Zeid, M. R., "Piece Part Coding And The Optimization of Process Planning', Proceedings - North American Metal Working Research Conference, May 1975.

16. El-Midany, T. T., & Davis, B. J., 'AUTOCAP - A dialog System For Planning The Sequence Of Operations For Turning Components', International Journal Of Machine Tool Research, Vol. 21, # 3/4, 1981, PP. 175-191.

17. Emerson, C., & HAM, I., 'An Automated Coding And Process Planning System Using A DEC PDP-10', Computing And Industrial Engineering, Vol. 6, # 2, 1982, PP. 159-168.

18. Eshel, G., Barash, M., & Chang, Tien-Chien, 'A Rule Based System For Automatic Generation Of Process Outlines For Deep Processes', Proceedings - ASME Winter Annual Meeting, Miami Beach, Florida, Nov. 1985, PP. 1-18.

19. Eskiciouglu, H. & Davies, B. J., 'An Interactive Process Planning System For Prismatic Parts', International Journal Of Machine Tool Research, Vol. 21, # 3/4, 1981, PP. 193-206.

20. Ferreira, P., Puls, F., Chang, Tien-Chien, & Liu, C. R., 'A CAD/CAM Systems For Complex Surfaces', Automated Process Planning Systems, ASME Winter Annual Meeting, Miami Beach, Florida, Nov. 1985, PP. 189-199.

21. Foster, Lowell W., 'GEO-METRICS II', Addison-Wesley Publishing Company, Inc., California, 1986.

22. Foston, Arthur L., Smith, Carolena L., & Au Tony, "Fundamentals of Computer Integrated Manufacturing', Prentice Hall, New Jersey, 1991.

23. Groover, Mikell P. 'Automation, Production Systems and Computer-Integrated Manufacturing', Prentice Hall, New Jersey, 1987.

24. Groover, Mikell P., & Zimmers, Emory W. Jr., 'CAD/CAM, Computer Aided Design and Manufacturing', Prentice Hall, New Jersey, 1984.

25. Halevi, G., 'The Role of Computers in Manufacturing Processes', John Wiley & Sons, New York. 1980.

26. Ham, I., 'Group Technology', Handbook of Industrial Engineering, Edited by G. Salvendy, John Wiley & Sons, New York. 1982.

27. Hannam, R. G., & Plummer, J. C. S., 'Capturing Production Engineering Practice within a CAD/CAM System', International Journal of Production Research, Vol. 22, #. 2, 1984, PP. 267-280.

28. Hegland, D. E., 'Out in Front with CAD/CAM at Lockheed - Georgia', Production Engineering, Vol. 28, # 11, 1981, PP. 45-48.

29. Hutzel, A., 'Integrating CAD/CAM through Group Technology', Proceedings Of the Annual Meeting and Technical Conference of The Numerical Control Society, Dallas, Texas, May 1981, PP. 430-444.

30. Hsu-Pin Wang & Wysk, Richard A., 'Micro Computer Based Process Planning Systems', Proceedings - Fall IIE Conference, December 1985, PP. 145-154.

31. Hummel, K. E., ' An Expert Machine Tool Planner', Computers in Engineering, Vol. 2, 1985, PP. 367-373.

32. ITT-Shop Practice Manual, P 9938, ITT Defence Communications Division, New Jersey, 1968.

33. Link, C. H., 'CAPP - CAM - I Automated Process Planning System', Proceedings - Annual Meeting & Technical Conference of The Numerical Control Society, Cincinnati, Ohio, 1976, PP. 401-408.

33. Meenakshi Sundaram, R., & Ta Jen Cheng, 'Micro Computer Based Process Planning Using Geometric Programming', International Journal of Production Research, Vol. 24, # 1, 1986, PP. 119-127.

34. Nau, D. S., & Chang, Tien-Chien, 'A Knowledge-Based Approach To Generative Process Planning', Automated Process Planning Systems, Proceedings ASME Winter Annual Meeting, Miami Beach, Florida, November 1985, PP. 65-72.

35. Nof, S. Y., & Barash, M. M., 'Dynamic Process Selection Procedures and Their Effect On Machine Configuration', International Journal of Machine Tool Research, Vol. 20, PP. 137-146.

36. Opitz, H., 'A Classification Scheme To Describe Work Pieces', Pergamon Press, Elmsfort, New York, 1970.

37. Philips, R. H., 'A Computerized {Process Planning System Based Upon Component Classification', Doctoral Dissertation, Purdue University, West Lafayette, Indiana, 1978.

38. Powers, John H. Jr, 'Computer-Automated Manufacturing', Mcgraw-Hill Book Company, New York, 1987.

39. Rebholz, M., 'Computer Aided Planning In Cold Forging', CIRP Annual, Vol. 29, # 1, 1980, PP. 173-177.

40. Riggs, James L., 'Production Systems: Planning Analysis and Control', John Wiley & Sons, New York, 1987.

41. Schaffer, G., 'GT Via Automated Process Planning', American Machinist, May 1980, PP. 119-122.

42. Schaffer, G., 'Implementing CIM', American Machinist, Special Report, # 736, August 1989, PP. 151-174.

43. Smith, R. N., 'Computer Aided Fully Generative Process Planning', Manufacturing Engineering, May 1981, PP. 98-99.

44. Steudel, H. J., 'Computer Aided Process Planning, Past, Present, Future', International Journal of Production Research, Vol. 22, # 1984, PP. 253-266.

45. Tirupatikumara, S., Joshi, S., Moodie, C. L., Kashyap, R. L., & Chang, Tien-Chien, 'Expert Systems In Industrial Engineering', International Journal of Production Research, Vol. 24.

46. Tulkoff, J., 'Lockheed's GENPLAN', Proceedings of The Annual Meeting and Technical Conference of The Numerical Control Society, Dallas, Texas, May 1981, PP. 41; 7-421.

47. Vogel, S. A., 'Use Of GT In Process Planning', Computers in Manufacturing Systems, ASME Planning, Control, and Robots, Presented at The Winter Meeting of The ASME, Chicago, II. 1980.

48. Vogel, S. A., & Adlard, E. J., 'The Autoplan Process Planning System', Proceedings of The Annual Meeting and Technical Conference of The Numerical Control Society, 1981, PP. 442-429.

49. Weill, R., Spur, G., & Evershan, W., 'Survey of Computer Aided Process Planning Systems', CIRP Annals, Vol. 31, # 2, 1982, PP. 539-551.

50. Wolfe, P. M., 'Automatic Process Planning Using Artificial Intelligence', Proceedings-Annual IIE Conference, May 1984, PP. 387-395.

51. Wysk, Richard A., Barash, N. M., & Moodie, C. M., 'Unit Machining Operations: An Automated Process Planning and Selection Program', Journal of Engineering For Industry, Vol. 102, November 1980, PP. 297-302.

52. Wysk, Richard A., 'An Automated Process Planning and Selection Program: APAS', Doctoral Dissertation, Purdue University, West Lafayette, Indiana 1977.

53. Zhang, S., Gao, W. D., 'TOJICAP-A System of Computer Aided Process Planning System', CIRP Annals, Vol. 33, # 1 1984, PP. 299-301.