Optimization of a Knowledge Based Design System Using AHP & Pattern Recognition (Case Study – Transformer Bushing)

A Thesis report submitted
In partial fulfillment of the requirement for
the award of degree of

MASTER OF ENGINEERING
IN
CAD/CAM & ROBOTICS

Submitted By
Shivpratap Singh
Roll No. 8048122

Under the supervision of

Mr. Gaurav Bartarya
Lecturer, M.E.D., T.I.E.T.

Mr. Shiv Prasad Subudhi
Consultant, TCS Limited

Mechanical Engineering Department
Thapar Institute of Engineering and Technology
(Deemed University)
Patiala-147004 (INDIA)

May - 2006

CERTIFICATE
2006

This is to certify that the thesis titled, **"Optimization of A Knowledge Based Design System Using AHP & Pattern Recognition", (Case Study: Transformer Bushing)** is submitted by **Mr. Shivpratap Singh,** in partial fulfillment of the requirement for the award of **Master of Engineering (CAD/CAM & Robotics) at Thapar Institute of Engineering & Technology (Deemed University), Patiala** is an authentic record of the initial work carried out by him. No part of this thesis has been submitted for the award of any other degree.

(Mr. Gaurav Bartarya)
Lecturer, M.E.D.,
Thapar Institute of Engg. & Tech.,
Patiala-147004 (Punjab)

(Mr. Shiv Prasad Subudhi)
Consultant,
Tata Consultancy Services Limited
Noida – 201301

(Dr. S. K. Mohapatra)
Professor & Head, M.E.D.,
Thapar Institute of Engg. & Tech.,
Patiala-147004 (Punjab)

(Dr. T. P. Singh)
Dean Academic Affairs
Thapar Institute of Engg. & Tech.,
Patiala-147004 (Punjab)

ACKNOWLEDGEMENT

Working without proper guidance and expecting success is just like making castles in the air, so whenever one wants to start any work, he/she requires guidance from experts. Having reaching to the brim of my thesis, I would like to express my deepest gratitude to all those cooperation enabled to me, to make this thesis a success.

I express my sincere gratitude to my guides, Mr. Shiv Prasad Subudhi, Consultant, Tata Consultancy Services Limited (TCSL), Mr. Gaurav Bhartraya, Lecturer, Mechanical Engineering Department, Thapar Institute of Engineering & Technology, Patiala, Mr. Prem Prakash Kalra (HOD, CSE –Deptt.,IIT Delhi) for their invaluable guidance, proper advice, painstaking and constant encouragement during the course of my work on this thesis.

I am greatly thankful to Mr. Sunil Kumar, I.T. Analyst, TCSL, and Mr. Anant Gupta, Assistant Consultant, TCSL, who devoted their valuable time and helped me in all possible ways towards successful completion of my thesis work. I am also thankful to Mr. Mayank Tyagi, I.T. Analyst, TCSL, for his inspiration and moral support which help me in completing the thesis.

I would also like to thank my parents, brother and sisters for their constant encouragement during the entire course of my thesis work. I would also like to thank all my friends especially Mr. V.Subramanyam, Mr.Ashutosh Chauhan, Manish Sharma and Miss.Poonam for their kind support as and when needed during the completion of my thesis.

I would like to thank entire staff of TCSL and M. E. D., T.I.E.T., Patiala, for their invaluable help.

Place: T.I.E.T, Patiala Shivpratap Singh

Date:

ABSTRACT

Present work addresses the problem of identifying the existing parts that, are similar, in one or many characteristics, to a new part at the design stage .The environment is UG/NX (3.0) where manufacturing partners share product related data to come up with new, customized, high quality products within a minimal time period or we can say with minimal lead times. The proposed method is based on the principles of Group Technology, Pattern Recognition and on the definition of Similarity of critical design attributes. A two step procedure is proposed: (First) A search procedure, which acquires and processes the search intent to retrieve the similar parts. (Second) A sorting procedure, which ranks these parts on the basis of their similarity using Similarity Index to the previously teach patterns. The first one is based on Binary Conversion and template matching using Analytical Hierarchy Process while the other one uses the ranking criteria based on Comparative index. The approach employs a systematic procedure to identify the closest similarity once the weights has been modified and generated for a particular file. A software system using C and Object Oriented Technology has been developed for Pattern Recognition using AHP concept. A Neural Net code based on Feed Forward Network for Pattern recognition is also developed to compare the capability of the AHP in the field of Group Technology. Same inputs were given to Neural Network code and the outputs were compared with those from AHP. Finally the nearness of outputs is compared for the weighted samples.

This dissertation goes with those,
who believe in dream of, and pursue incessantly in the progress of the world!

INDEX

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF GRAPHS

LIST OF ABBREVIATIONS

1. **ADS –** Automated Design Systems
2. **AHP-** Analytical Hierarchy Process
3. **AI** – Artificial Intelligence
4. **API –** Application Programming Interface
5. **ANN –** Artificial Neural Network
6. **BOM –** Bill of Material
7. **CI –** Consistency Index
8. **CAD –** Computer Aided Design
9. **CAM –** Computer Aided Manufacturing
10. **CAE** – Computer Aided Engineering
11. **CBR –** Case Based Reasoning
12. **CBED/MC –** Case Based Evolutionary Design for Mass Communication
13. **COM –** Component object Modeling
14. **CAAD –** Computer Aided Architectural Design
15. **CBD –** Case Based Design
16. **CIMS –** Computer Integrated Manufacturing Systems
17. **CBC –** Component Based Classification
18. **CR –** Consistency Ratio
19. **DITL –** Design in the Large
20. **DITS –** Design in the small
21. **DCOM –** Distributed Component Object Modeling
22. **DMC** – Decision Making Criteria
23. **DM –** Decision Making
24. **ES –** Engineering Services
25. **ED –** Engineering design
26. **GT –** Group Technology
27. **GM –** Geometric Mean
28. **GRIP –** Graphical user interface
29. **JIT –** Just in Time
30. **KBE –** Knowledge Based Engineering
31. **KBS –** Knowledge Based Systems
32. **KEE –** Knowledge Enabled Engineering

33. KF – Knowledge Fusion

34. MCDM – Multi Criteria Decision Making

35. MADM – Multi Attribute Decision Making

36. OCR – Optical Character Reader

37.OOP – Object oriented Programming

38. PFA – Product Family Architecture

39. PR – Pattern Recognition

40. RI – Random index

41. SI – Similarity Index

42. SLCA – Single Linkage Crystal Analysis

43.SRS – System Requirement Study

44. UG/NX – Uni - Graphics

45. VDU – Visual display Unit

46. 2 D – Two Dimensional

47. 3 D – Three Dimensional

CHAPTER- 1

DEISGN & AUTOMATION

(An Introduction)

1.1. Introduction [23]

The design engineer's dilemma is to develop improved and less expensive products that can be manufactured in less time. The demands are the challenge for every design engineer, In order to achieve this product development process has to be refined and improved.

This report will give an introduction to knowledge based engineering (KBE) in product development. KBE can be defined as a tool for acquiring knowledge and reusing it. The concept of KBE is therefore very broad .a spread sheet is a tool for capturing knowledge by implementing equations ,or rules which enables knowledge recycling. Thus the focus in this report is firstly KBE tools where knowledge is stored as objects which belongs to different classes in an object oriented way like Java, C++ among others .Secondly these are KBE tools which are often coupled to a geometric engine to enable automatic generation of product concepts in terms of virtual prototyping.

KBE is a subset of Knowledge Based systems (KBS) which could be clad as a spin off from Artificial intelligence AI.KBS is often refers to as "expert system" because they intend to capture expert's knowledge and sometime also generate creative solutions .KBE on other hand is used to automate mundane time demanding tasks. By freeing people from this routine work, more time could be used to come up with new innovative solutions.

Benefits with KBE are that the optimization of product concept is easier and product and process knowledge is stored .Drawbacks are that, it is time demanding to develop KBE systems and it can sometimes be seen as a "black –box" if the automated procedures are not clearly explained to the users.

Today's research is focused on improving the development methodologies to enable shorter development time and improve the quality of the systems .KBE is economically demanding to implement in a company structure while mostly large enterprises employ the tool. Therefore it is of interest to develop methods for KBE for small and medium size enterprises (SME's). Initially KBE in product development means generating a product concept. Now the capabilities of integrating the other aspects of other processes are explored. One of these is the manufacturing processes and the possibility for concept evaluation in terms of manufacturability.

1.2. Research Paradigms [33]

Product development drives business growth but at the same time, it is expensive. The industry average is one product success for every three thousand ideas. A structured and reliable approach is required in engineering to ensure high return on the innovation investment.

Transformers are custom designed products. Practically every order requires a separate design and new set of drawings. Though certain assemblies/components have been standardized, these constitute a small part of the total design effort. A typical transformer may consist of as many as 3000 components/sub-assemblies, and the complete engineering information requires 500 drawings/Bill of material. The effect of this huge design/drawing work is that the issue of engineering information takes as much as 25% of the total delivery cycle of the transformer.

All the components /sub-assemblies/assemblies that go into a transformer have to be so designed that they can be assembled properly; any mismatch identified at the time of manufacture is difficult to rectify. In addition, all modifications carry a certain penalty in terms of the cost of the product as well as the time taken to manufacture. Since a large number of drawings have to be made, they are made simultaneously, adding to the chances of mismatch.
Any engineering drawing has basically, two sets of dimensions that are most important part of drawing. These are
a. Dimensions which are independent variables, i.e. they are derived from the specifications, ratings, customers requirement, etc

b. Dimensions which are dependent i.e. they are derived from either the independent dimensions- based on certain formulae or are derived from certain standard practices or design rules.

Using this logic, it is possible, if the independent dimensions are available, to work out the dependent dimensions, and thus all the dimensions needed to make the drawing can be worked out.

If we take a top down view of the drawing process, we will see that an assembly consists of a number of components. Certain dimensions derived from the assembly are common to a number of components. When the drawings are being made, any mistake in the common dimensions, either for assembly or component drawing would create a mismatch. In the ordinary courses, all the drawings with common dimensions should be checked at the same time to ensure correctness. The fact that all such drawings may not be available, as well as the possibility of human error creates mismatch that have to be rectified later at a cost to the organization.

1.3. Research Purpose

The actual product ownership remains with the manufacturer developing the functional products. Controlling the life cycle cost of the product is a critical issue. The life cycle of a product is greatly decided upon in the conceptual phase where few support tools exist [3]. It is difficult or even impossible to compensate for a poor conceptual design when the detailed design phase starts, [30]. As functional products include the development of both hardware and accompanying services, people from all areas and their knowledge will be needed in the decision-making. There is somehow a need for support systems that can perform activities fast and access the knowledge earlier.

1.4. Design Automation scope in transformer Automation: [33]

All manufacturing business agreements are being made on a life cycle basis where the actual product ownership often remains with the manufacturer. Designing the actual life cycle of the product is now of great interest. To design the life cycle of a product, understanding how the decisions made during the design process will form the life cycle properties is vital. Could the life cycle of a functional product perhaps be simulated in the conceptual phase and thereby help the engineers to design it?

A question has been formulated in order to define the scope of this research:

"How productivity of transformer industry can be enhanced in the concept phase of the product development process **by means of knowledge based engineering**?"

A new approach has been taken to maximize the productivity. Computer programs have been developed, which take the dimensions identified as an independent variables as inputs. The logic for calculating dependent variables is built into program, and dependent variables calculated. Based on these dimensions and the relations between them, a 3-D model of the Component/assembly is generated by the program. Finally, the program also generates the 2-D product drawings. Since separate programs would cover different assemblies, all the common dimensions are available while making the component drawings. The chances of errors/mismatch are thus reduced. Since all the data required making the Bill of material is available, the program generates the bill of material also. The advantages of such a system are:

- By giving one set of data for an assembly, a number of drawings and the Bill of material would be generated. Thus the time taken to make the drawings and the bill of material is reduced.
- Since the common dimensions would be given only once, errors due to mismatch is reduced.
- As a 3-D model would be available, the designers can rotate the model and visualizes it from different angles and thus make improvements.
- Fouling between different components /assemblies, excess clearance would be visible on the computer screen in 3-D, allowing correctness to be before the drawings are released, or components procured.
- The bill of material would be generated by the program, and can be stored in a data base for use by the planning groups.
- Since all the changes are built in the program, there would be commonality of logic. Mistakes due to adopting the wrong formulae by an individual are therefore eliminated.
- Any changes in the design philosophy would entail modifications in the program and need to ensure proper changes in the individual designs would be elminated.This would ensure deter design control.

Based on above philosophy, an Automated Design System (ADS) with the help of knowledge based engineering (KBE) would be developed. The system is suitable for the design of the transformers following any transformer manufacturer organization's design methods. The system would be very comprehensive and generate analyzable model and assembly, drawings and bill of material for a large variety of transformers with different ratings.

The system would be designed with menu driven prompts so that a user without any knowledge of 3-D modeling software or database package can use the system. The only need is that the user should know how to use this system. The basic modeling software used to develop this system is Unigraphics due to its strong knowledge based approach. The menu screens for making the system user friendly as well as the calculations etc has been developed in Knowledge fusion and Ufunc.

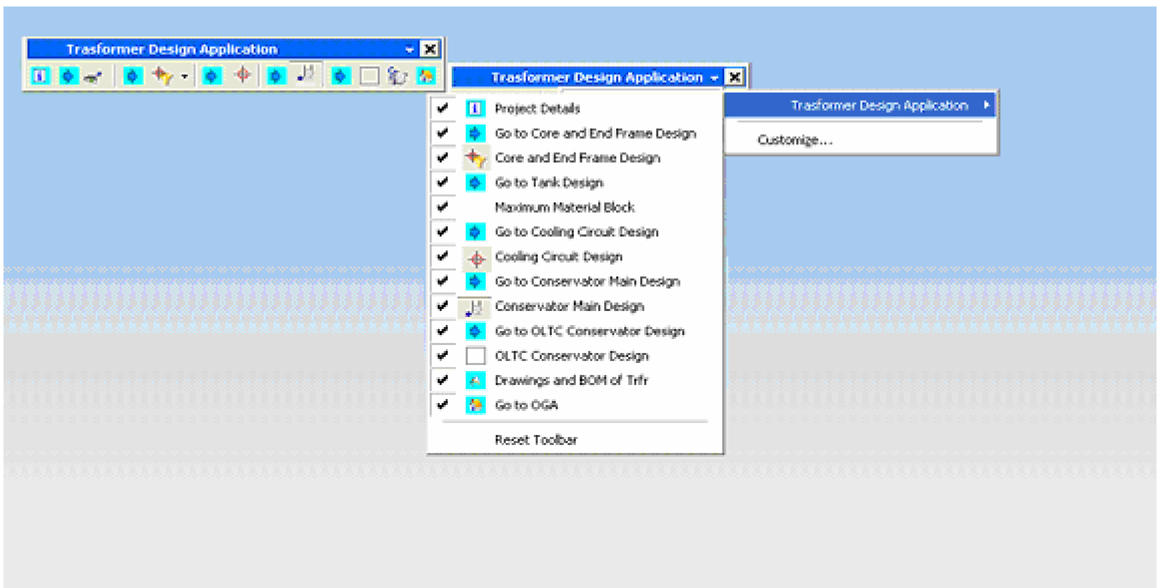


Fig: 1.1. Showing the GUI been prepared in the software

1.5. AIM:

In the present work the aim to develop a KBS using KBE for design and automation of a transformer. Here the first requirement is to select a tool using which KBE would be initialized for Automation of transformer.

A large number of tools are there in this domain varying in capabilities as per the application requirements and demand. The most promising one in the existing scenario, that are widely used in today's industry are (Proe, Catia, UG/NX, Solid works and Cam pro Technologies). A fruitful survey from [1, 2, 3, 4, 5] shows that UG could handle most of the problems and logics modeled in SRS.

1.6. Tools selection to develop KBS: [28]

Industry pressure to reduce cost and improve quality is driving growth in the use of digital simulation through out the life cycle. Choosing the right tool is the key to achieving the business benefits. Market survey shows that UG/NX3.0 is the most supported KBE tool in the industry.

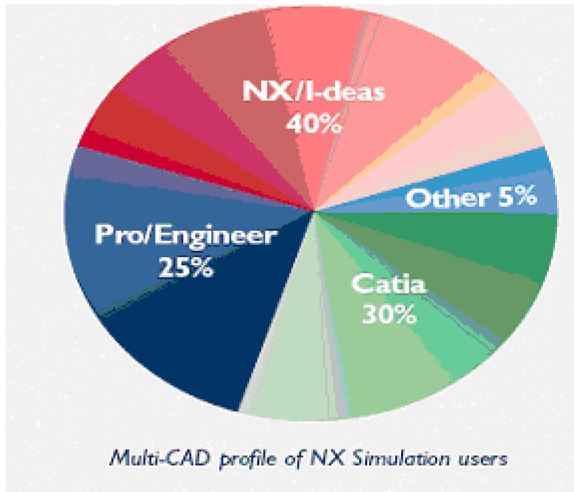


Fig: 1.2 showing potential of UG in Market [28]

1.7. Why UGS: [28]

It helps manufacturers to develop right product at the first time, with a complete range of simulation, validation and optimization tools. NX product simulation applications include dynamic motion simulation, basic strength analysis, system-level performance evaluation and advanced response, durability, fluid flow and multi-physics engineering analysis for robust simulation of functional performance.

NX also manages CAE processes, workflows and data to deliver vital performance feedback where it is needed to improve your products. One more relevant clue in favor of UG is: "Simulation is transforming from a reactive and defensive tool to a proactively deployed competitive weapon."(Daratech, Inc. Industry update, March, 2004).

A key strength of the NX environment is its capability to incrementally capture expert knowledge to automate simulation processes. Companies can harness knowledge with the NX Simulation Process Studio to standardize best practices and analysis procedures for use by less expert users. Companies can implement from the bottom up to help eliminate cultural barriers, reducing risks and increase effectiveness of the resulting simulation environment.

UGS technologies uniquely support Lean Design initiatives, helping companies reduce delays, maximize design re-use, reduce defects, and improve process efficiency. The primary solution components enabling Lean Design are NX Digital Product Development and Team center product suites. Together they provide a managed development environment; a unified solution that addresses all phase of product development, from concept through manufacturing; knowledge driven automation; simulation, validation, and optimization; and system-based modeling. All of these tools and technologies can be directly applied to eliminating waste and successfully implementing lean development initiatives.

1.7.1. Product Development in UG: [37]

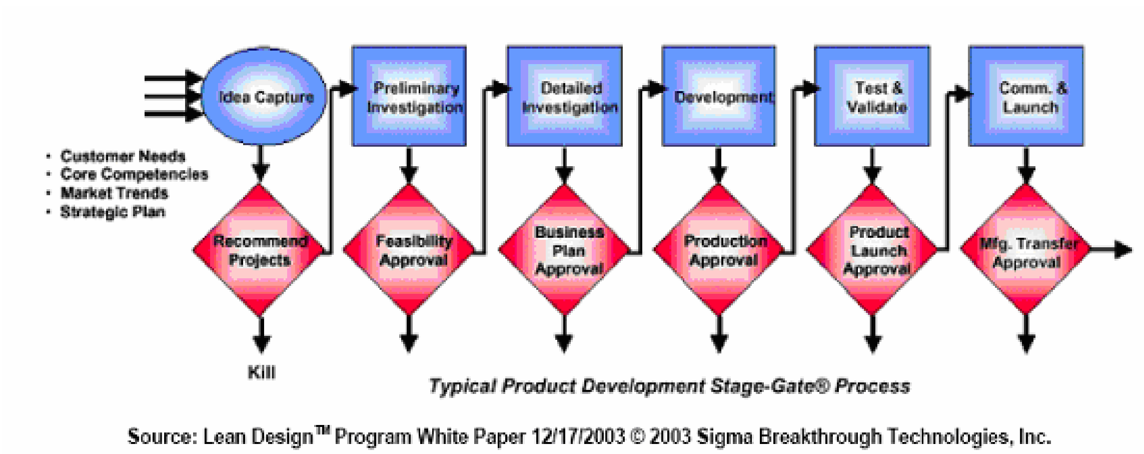


Fig: 1.3 Product development Stage process [37]

Lean and the Six Sigma methodologies applied primarily to manufacturing operations have proven that it is possible to achieve dramatic improvements in cost, quality, and time by focusing on process performance. Whereas Six Sigma is most closely associated with defects and quality, Lean is linked to speed, efficiency, and waste. The essence of Lean is to concentrate effort on removing waste while improving operations. The simplified notion of Lean Product Development/Lean Design is to remove waste from all aspects of the product and associated development process before it ever gets to the manufacturing floor. The term is derived from *lean manufacturing*: continually reducing waste in all areas and in all forms. It is an English phrase coined to summarize Japanese manufacturing techniques including:

KANBAN: only make what is needed for the next station. (Reduce bottlenecks, minimum inventory).

Just -in-time: Bring materials from suppliers "just-in-time" to assemble them into the product.

To be cost competitive, front-line staff must learn to see waste and relentlessly pursue its elimination not only on the manufacturing floor but also on their engineering drafting boards and computer screens.

1.7.2. Opportunities for reduction of waste in UG for Design: [37]

The areas of waste in manufacturing, which are relatively easy to identify and quantify, can be extrapolated to apply to other product development processes. The following table identifies and evaluates waste in product design, summarizing the findings of two studies.

Opportunities for reduction of waste in design		
Area of waste reduction	Percentage of design waste	
Designs never used, completed, or delivered	Unknown	
Downtime while finding information, waiting for test results, etc.	33-50%	
Unnecessary documents and prototypes		
Underutilization of design knowledge, for example in costly parts	18%	58%
Over design, such as features customers don't need	8%	
Validating manufacturing errors early in the design process	17%	
Poor designs producing product defects	15%	

Sources: UGS analysis of Tier 1 Automotive suppliers; A.T. Kearney, *The Line on Design*, 2003.

Fig: 1.4 Opportunities for reduction of waste in Design

1.7.3. Benefits of UG: in reducing waste in product development [37]

The basics principles of lean manufacturing can be applied to product development as well. The categories of waste in manufacturing can be mapped to associated areas targeted for waste reduction in product development.

Lean Manufacturing: The 7 Deadly Wastes	Applied to Lean Design
Delays Movement and transport	**Reduce Delays** • Finding information, waiting on test results • Unnecessary documents, physical prototypes
Excess production and early production Inventory	**Maximize Design Re-use** • Learning from past design experiences • Reducing unnecessary features • Designs never used, completed or delivered
Poor process design Inefficient performance of process	**Improve Process Efficiency** • Underutilization of design knowledge, for example in costly parts • Early validation of manufacturing errors
Making defective items	**Reduce Defects** • Poor designs • Warranty issues

Sources: UGS analysis of Tier 1 Automotive suppliers; A.T. Kearney, *The Line on Design*, 2003.

Fig: 1.5 Benefits of UG in reducing waste in Design

So, the above survey shows that UG could reduce the lean time for the product development up to great extent and could help in achieving the production and development goals with maximum benefit. UG equipped its users with large number of tools for automation which are as follows:

(1) knowledge driven Automation
(2) system Based Modelling
(3) integrated Collabration
(4) Open by design
(5) Product proven applications

In present work only knowledge driven automation is considered as a base tool.

1.8. Knowledge Driven Automation [38]

Knowledge Fusion is the underlying technology that both allows many of the Process Wizards to be built, and allows customers to capture their own knowledge directly into their product designs. In Unigraphics NX, several steps have been taken to further expose this powerful technology to the average user. Because Knowledge Fusion allows the maximum flexibility to the customer, it enables them to capture their own best practices or competitive advantages. Additionally, Knowledge Fusion enables expertise that may be in the heads of the most experienced people in the company to be captured and replicated by the least experienced users.

This not only enhances the automation processing but provides a power tool to the user for decision making. Solutions integrate knowledge and optimize engineering processes across the product lifecycle more completely to create better products through digital decision-making. While other CAD systems focus on form and fit in product design, Unigraphics NX adds the ability to capture the function the product is intended to perform.

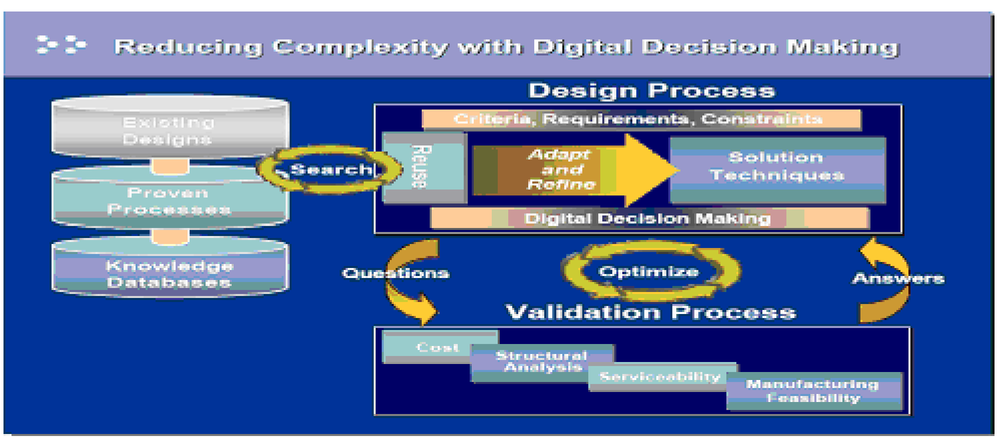


Fig: 1.6 process graph for digital decision making [38]

So in broad sense it would work as an Expert system which could retrieve data from the database, as per the information provided by the user and could update the required dimension or material properties . This logic also works visa versa on the same lines, i.e. it could produce BOM and log file for the changes made in the database on the user information.

1.9. Knowledge-enabled simulation

With NX knowledge-enabled capabilities, companies can create templates that can be instantly loaded and automatically executed as stored processes. As an example, a wheel manufacturer can capture its best practices for designing and analyzing various types of standard wheels and define that process in a template file. They need only drag that template onto the geometry and start the process. These same automation techniques enable preferred simulation processes to be followed for each type of workflow or product evaluation activity leading to higher staff productivity, higher product quality and consistency of engineering results. This effectively brings fundamental performance simulation activity into the upfront design process and supports enterprise wide initiatives to capture in-house knowledge and proven repeatable simulation methods.

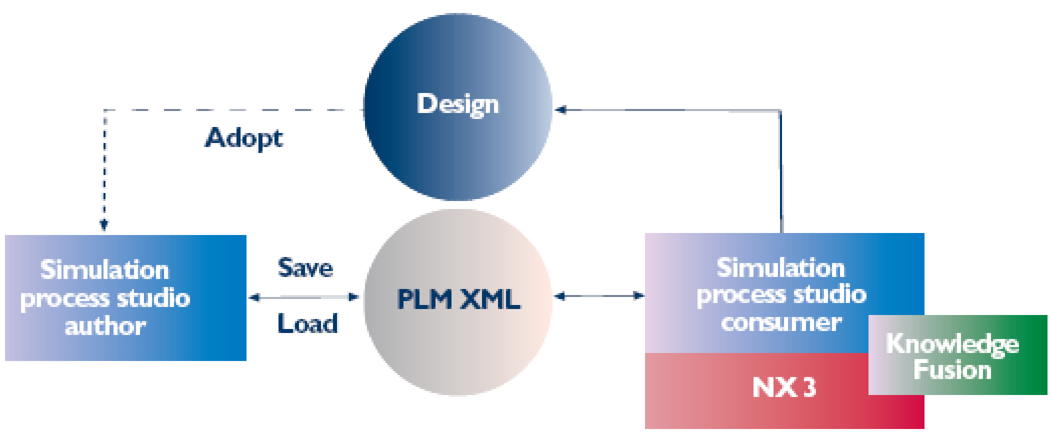


Fig: 1.7 knowledge enabled simulation flowchart [28]

1.10. Automation of Transformer using KBE:

Automation could be defined as **"A transformation of manual processes to a machine dependent feedback system".** If considering this as the base terminology than with respect to CAD , conversion of engineering drawing i.e. hand made to a machine dependent (i.e. Computer) feedback system (i.e. using Knowledge based engineering (KBE) in UG) could be considered as a perfect example of Automation.

Automation in industrial and manufacturing industry can be segmented as follows:

(1) Integration of electronics and mechanical systems (Mechatronics)

(2) Automated design and drafting (processing)

(3) Automated identification and retrieval system (inventory)

(4) Automation of shop floor with unmanned vehicles and manufacturing processes.

"Mechanical Automation group concerned with the design and development of methods and equipment for the control and automation of mechanical systems and physical processes." The definition above by "K.L.Moore" gives a clear view that automation is a tool ,which could support industries in achieving their time bound goals , increasing rate of production , reducing cost and last but not the least providing stability and strength to the organization to remain competitive in the present scenario.

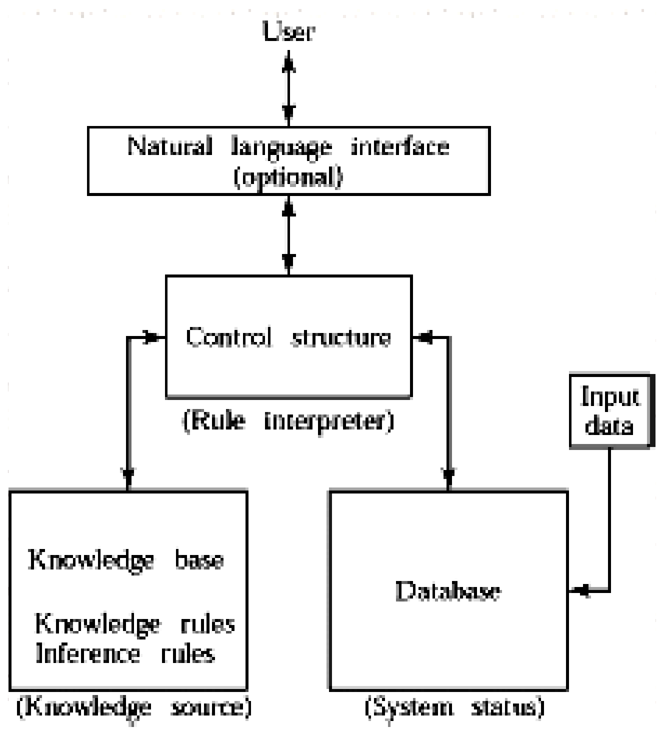


Fig: 1.8 Flowchart showing automation using KBS [21]

The above claim of automation can be supported by providing some facts and figures

1.10.1. Example: 1 (Motorola – "Business India", December 2003)

Motorola group when entered in Indian market have a capability of producing 3000 sets per day, using the ordinary methods that exists in the market but that was not sufficient to compete their rival group whose production rate was 10000 sets per day. The problem was clear they need a higher production rate. The first step was to trace out lacuna in the process.

The researcher shows that their main problem was that in industry a chain process is followed i.e. Hierarchy process. This could be explained as follows let (1) a new part is designed by R&D peoples (2) next it would be passed for analysis (3) than it would be transferred to shop floor and then they would provide the feedback that technology does not allow the miniaturization of chip of chip to 3x10^-6mm and the whole process to be repeated.

After the survey organization decided to apply Concurrent Engineering, that during designing a new component call, all the departments to be called on a common platform, and once the part has been approved all the departments start working parallely. This increase the production rate to 20000 hands sets per day.

1.11. Summary of the application to be designed:

The example proves the capability of automation to remain competent in the market. Following the law of "Survival for the fittest" automation in industry has become need of the day. That was the case in 2003 and now we have been equipped with much advanced technologies like KBE, KBS, Expert system, Case based reasoning, agile manufacturing. In the present work **"automation a Transformer designing process", its drawing (i.e. conversion of all 2D drawings to 3D) and all sorts of parameters related to its manufacturing** using Uni-Graphics with **knowledge base engineering**. In short it could defined as a type of Knowledge base system which would be based on large no of logics, all sort of knowledge hired from tech brains of the industry, all sort of database required to update the model and a large number of options to explore the the performance of transformer on varying the one or other parameter.

1.12. Brief overview of transformers: [33]

High-power flow in a transformer is encountered during peak hours of operation whilst a smaller amount of power flow is found during the off peak hours. This variation in load flow must be taken into account when designing a transformer. In addition to changing load flows, transformers must be able to cope with various fault conditions. These fault conditions are simulated prior to the transformer being commissioned. The relevant Australian Standard states that the three tests to be conducted are the Separate Source Voltage Withstand Test, the Induced over Voltage Withstand test and the Lightning Impulse Test.

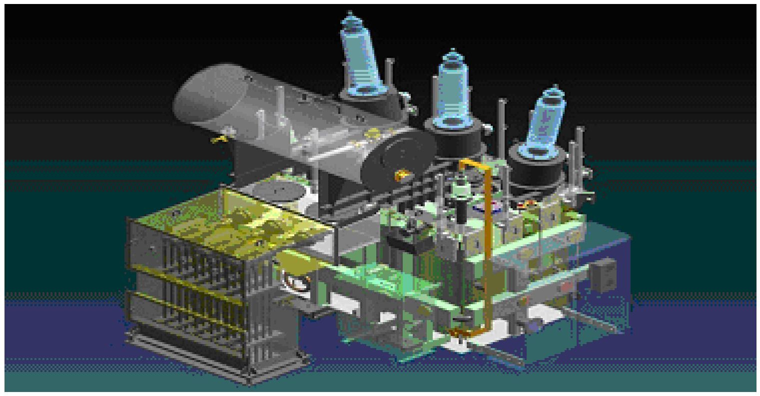


Fig: 1.9 Electric Transformers [33]

1.13. **Design Automation**

During brain storming session we found that a transformer merely consists of 3000 components. As per the SRS (system requirement study) the whole modeling is purged under 93 models. Whole of the work is divided into three modules (1) Modeling with KBE (2) Integration with database (3) Customization and automated generation of BOM taking values from the spread sheet.

1.14. General Requirements [SRS]

SRS provides us certain constraints keeping these into consideration the project outlines could be modeled .The general requirements are as follows:

(1) The Drawing Notes can be in a separate file, which can be incorporated in the required drawing and modified

(2) Where Drawing is generated. Drawing no will be asked by the user. If item/assy of this Drawing no is called in other assy drawing. then this drawing no will be called against that item no in the BOM

(3) For editing any Feature, there should be an editing icon, by which we can edit that and need not go back.

(4) Attributes to be specific to the variants called.

(5) Where ever by default is mentioned, the feature should be first created and it should have an EDIT Button, by clicking the EDIT Button the Corresponding Dialog should open to change the parameters of that feature.

(6) The Expressions should be in administrative control

(7) Variants will also be reported in the main BOM. Things which are standard may not be to exact detail but there attributes and BOM need to be accurate, Qty to be given in BOM

(8) Overall there are 105 models of bushing, but to model all of them will take lot of time. So a way is to be sorted out, so that just by teaching 5-6 patterns on the basis of similarity rest could be modeled by modifying the previous samples.

(9) Bushing rotation about its central axis and warning should be generated if user gives wrong input for the cable box to the system.

(10) a general requirement for all round pad is that they should not be placed in such a manner that a plane created by opposite holes on the pad should never be parallel to transformer center plane

Above shown are some most generalized requirements of the SRS, going in to the details it is found that each of them is a separate module and should be treated separately.

The problems which will be handled in present work is the modeling and designing of bushing along with the implementation of logics or we can say constraints defined in 8th and 9th Para of the SRS requirement.

1.15. Concluding Remarks:

To deal with the problem intricacies some advanced tools and technologies are required. These technologies will be used further in thesis to achieve the different goals and targets.

In the next phase a course over Group Technology, Analytical Hierarchy Process and Pattern recognition will be reviewed. These would be used in sequence to solve the optimization problem.

CHAPTER 2

KNOWLEDGE BASED ENGINEERING

(LITERATURE SURVEY)

Introduction: [24]

Today industry is facing the challenges to store their existing design, and use of knowledge in new component form existing design and help the customer to visualize the design in minimum time. New designs are evolved with implementing the knowledge of designer and new concept. Most of the designs are not reused after they are supplied to the customer.One of the existing solutions to reuse the existing designs and reduce the design time is Knowledge Based Engineering. Knowledge Based Engineering not only stores the design rules but it reduces the dependency on the designer also. It also helps in creation of 3D model and 2D drawing of the model to reduce modeling and detailing time.

The success of manufacturing companies depends on their ability to produce high-quality products at the lowest cost. This applies to a transformer industry that aims to create designs that are optimized for manufacture. In striving for lower cost, it may be worthwhile to focus on the concept phase of product development, since most of the product cost is committed there. This cost is, however, not often seen until it is allocated later or downstream in the product development process, for instance, during component manufacture. Currently, manufacturability assessment is often conducted manually in the transformer industry, which takes time and entails the risk of missing design flaws. Life-cycle modelling has become a practical method of visualizing this cost and enabling product analysis by simulating product development activities in design for manufacturing support tools. The aim of this work is to discover how knowledge enabled engineering (KEE), an approach which includes engineering design, knowledge based engineering (KBE) and similar knowledge intensive methods, can be used to improve manufacturability evaluation of transformer component.

2. 1 INTRODUCTION TO KNOWLEDGE ENGINEERING.

If systems engineering is the discipline of producing software and hardware solutions to users' information needs, then sub-disciplines can be defined to address development techniques designed to minimize identifiable risks. Software engineering provides the mechanisms for validating the implementation of well specified algorithms. Human Computer Interaction provides analysis and design techniques based on prototyping of the user interface to address aspects of systems where the risks are associated with the users' needs, or the system usability. Data engineering addresses the permanent storage of large amounts of data and the efficient retrieval of the relatively small portion required for any process. In contrast, knowledge engineering addresses the structure of complex but ill defined processes where the solution to defining the process is to define the knowledge involved in the process explicitly in a knowledge based system (KBS)

Conventional software development follows the waterfall life cycle model. Conventional software engineering approaches produce efficiently implemented code to execute algorithms to perform required functions which will always produce the correct outcome for correct input. The knowledge engineering approach allows users and experts to describe requirements and methods to perform the required functions at a high level close to the one in which they think about the task: the Knowledge Level. These can then be presented back to them for validation of the content, and modification.

Knowledge Engineering (the process of developing KBS) differs from conventional software engineering mainly at the early stages of the life cycle when user requirements and functional methods (or Knowledge) are being acquired. The tools for implementation, user interface design, testing, maintenance and updating systems may differ, but the principles which govern all software systems are the same. Therefore, although the early stages of knowledge acquisition will involve a knowledge engineer and a (or more) domain experts, later stages will involve software engineers for implementation/integration.

Since the expert's knowledge must be represented at the Knowledge Level rather than at lower computationally efficient levels during knowledge acquisition, the computational representation at high levels normally persists throughout a KBS life cycle. Therefore the delivery process for KBS is not the same as for normal software. It is generally constrained into a triptych architecture including a knowledge base containing a representation of the acquired knowledge, an inference engine which reasons over this knowledge and a data area to store the data from a particular "run" of the system. This allows the processing to be separated away from the knowledge and isolated in the inference engine, leaving the knowledge free of procedures below the knowledge level.

2.2. Background of KBE [24]

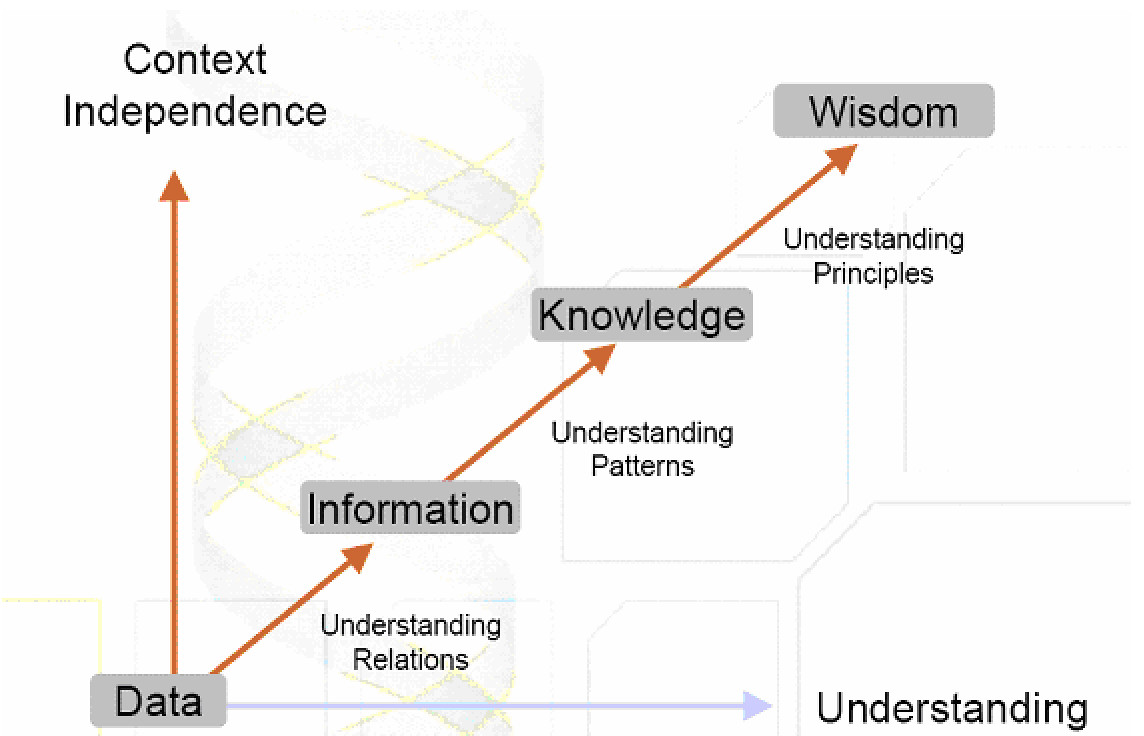


Fig: 2.1 representing the ladder step graph for Knowledge Based systems [24]

In the 70's design drawing table started to be changed for computers and products could be developed in a two-dimensional computer environment rather than on paper. This step made the modeling process easier and faster. Around ten year later the first solid modeling systems were employed i.e. CAE/CAD/CAM. This was the big improvement and gave a new era to the design work by enabling a better overview and is an important step into paradigm of virtual prototyping.

The next big step for the product development process was when all its player were started to be integrated in the beginning of 90's. Since then the integrated product development process has been improved in different ways. One of which is utilizing a tool named Knowledge Based Engineering. It is stated that KBE will have the same importance in the organizations in 2010 that the CAD/CAM/CAE had in 90s. The concept is not a new concept. On this the researches were performed in 50s with the objective of developing a system which had its own intelligence, known as Artificial Intelligence (AI). Today Knowledge Based system (KBS) or Expert Systems are closely related.

The key to success was to let the engineer do the creative work and use the computer to automate the routine work. In the beginning of 80s researcher started to store knowledge and rules in systems which then manage to perform mundane tasks of the product development process. The concept of KBE was born.

2.3. KBE Fundamentals [23]

The knowledge Based Engineering can be defined as:
"The use of advanced software technique to capture and reuse product and process Knowledge in an integrated way"
The various definitions of KBE are given by different researcher. Some of them are as follows:

- "Knowledge-Based Systems are a special class of computer programs that purport to perform, or assist humans in performing, specified intellectual tasks." "**Dixon**"

- "KBE systems aim to capture product and process information in such a way as to allow businesses to model engineering design processes, and then use the model to automate all or part of the process." "**Chapman**"

- "A Knowledge-Based System is the one that captures the expertise of individuals within a particular field, and incorporates it and makes it available within a computerized application. A KBE application is further specialized and typically has the following components: Geometry Configuration and Engineering Knowledge." "**Lovett"**
- "KBE is a technology that allows an engineer to create a product model based on rules that capture the methodology used to design, configure and assemble products. KBE facilitates the capture of the intent behind the product design by representing the why and how in addition to what of a design." "**Bailey**"

- "Knowledge Based Engineering is the execution of engineering tasks using knowledge that is not normally immediately accessible to the designer or engineer, and that has been purposefully accumulated and stored for use by the designer or engineer, usually (but not always) in some computer-mediated form. Thus, KBE usually (but not always) implies the use of some kind of computer system, examples of which include the so-called expert systems, web-based knowledge bases, and the like." "

"Permoyer & Burnett"

These are few definitions for the Knowledge Based Engineering.

The main objective of KBE is to reduce lead time by capturing product and process knowledge. The core of the system is the product model where the product and process knowledge is stored. The figure below shows a KBE system. External database are used for tabled data. Input to the KBE system is usually customer specifications, which in turn gives several kinds of outputs when being processed. The system software is object oriented and can therefore perform demand driven calculations.

A product can often be divided into several parts which contains the details. The parts are managed by an object oriented system that specifies the relationship to sub and super parts.

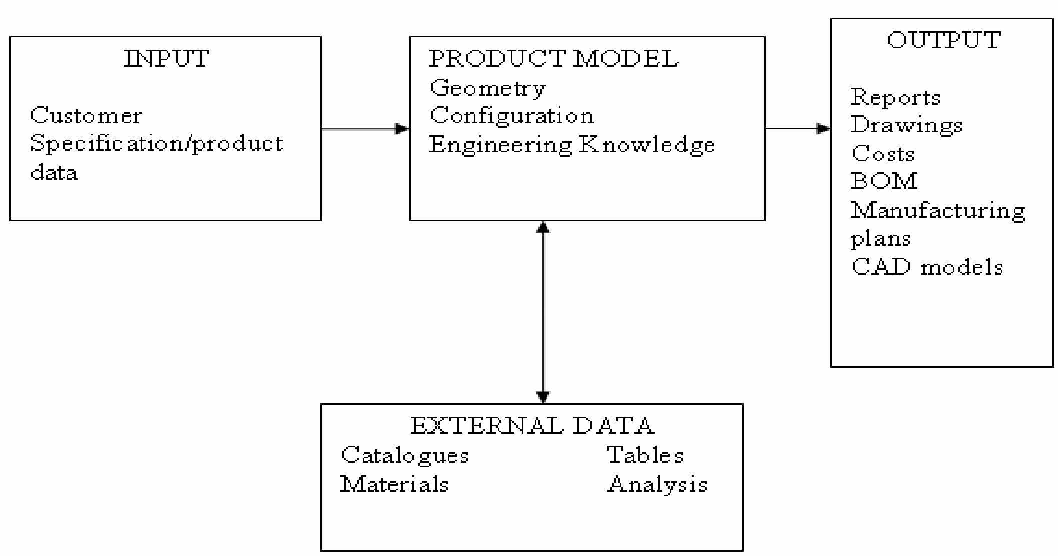


Fig: 2.2 showing block diagram of a KBE system [23]

2.4. Literature review of KBE

(a) MITCHELL M. TSENG and JIANXIN JIAO [25]: - As the natural technique to achieve mass customization, reuse can be best approached by Case-Based Design (CBR) paradigm. Product Family Architecture (PFA) provides a unifying product platform to cover diverse customer needs while keeping the economy of scale. In Case-Based Evolutionary Design for Mass Customization (CBED/MC), PFA plays an important role in acquiring highly-specialized domain knowledge and organizing previous design cases. With the introduction of the case class, case memory stores designs in inheritance and partonomic hierarchies, which enables cases to be represented in an object-oriented fashion. Case classes conform to the product classifications in PFA, while cases correspond to specific products. These class-member relationships facilitate the layered indexing and retrieval by index elaboration. The adaptation problem can thus be solved by opportune traversals through the set of conceptual classes. Based on the methodology presented, future efforts will be geared towards implementing a CBED/MC system.

(b) Craig B. Chapman and Martyn Pinfold [9]: - uses the knowledge based engineer system to extend the current capabilities in automotive Body-In-White (BIW) engineers. It allows them to respond dynamically to change with in rapid timeframes and to access the effect of change with respect to constraints imposed upon them by other product cycle factors. The system operates by creating a unified model description that queries rules as to the suitability of the concept design and built using standard KBES to reduce project cost and system implementation. Knowledge from expert engineers and technical literature are captured and represented with in the KBE application frame work. As the use of computer technology in the design and analysis of automobile has been developing at an increasing rate, so such system are currently piecemeal and provide solution to very specific problem. BIW in an automotive term is used to describe the structural body of the vehicle. The structure is compromised of many substructures that come together to dorm a framework. The basic concept structural modeller is shown below

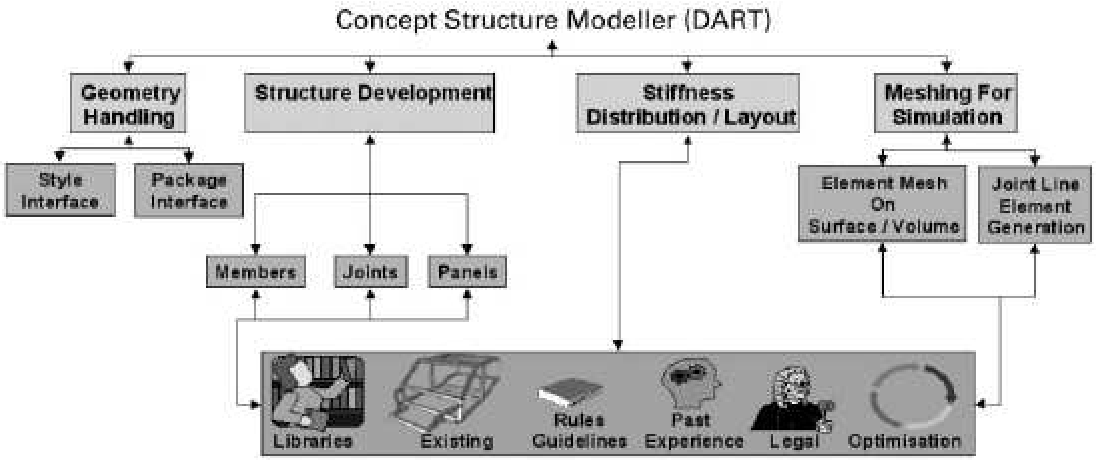


Fig: 2.3 Initial basic layout for Design Analysis Response Tool (DART) [9]

(c) Marcus Sandberg, Peter Åström, Tobias Larsson and Mats Näsström [31]
Knowledge based engineering (KBE) has emerged during recent decades as a popular way of supporting design tasks. It is commonly claimed that the benefits of KBE are greatest if the product change from one product in the product family to the next is minor. KBE is also preferably used for routine design tasks where a designer makes knowledge-based decisions on a daily basis. The increase in engineering productivity through the use of KBE results in tedious, time consuming, error prone and repetitive tasks being automated Using knowledge enabled engineering (KEE), a design tool connecting CAD and distortion assessment using FEA was developed. The design tool consists of CAD software coupled to finite element software (MSC.Marc™) by means of Python and UNIX scripts. The design tool is controlled through a graphical user interface.

(d) Patrik Boart, Petter Anderson, Bengt-Olof Elfstorm [7] uses the Knowledge based engineering technique to automate and provide system support for engineering activities, including decision making capabilities built on "engineering knowledge". Rules and data are updated as the KBE system evolves, both during the pre-study of the project and the emergence of the project final concept. Here the Knowledge enabled approach is used for the Finite element Analysis during the product development. In this much focus has been spent on both improving the accuracy and the robustness of the numerical predictability or the computing power itself. Due to automation he concluded that the large number of concept can be generated due to the KBE and large knowledge data leads to less processing time and increase in accuracy.

2.5. KBE System Development and Designing [26]

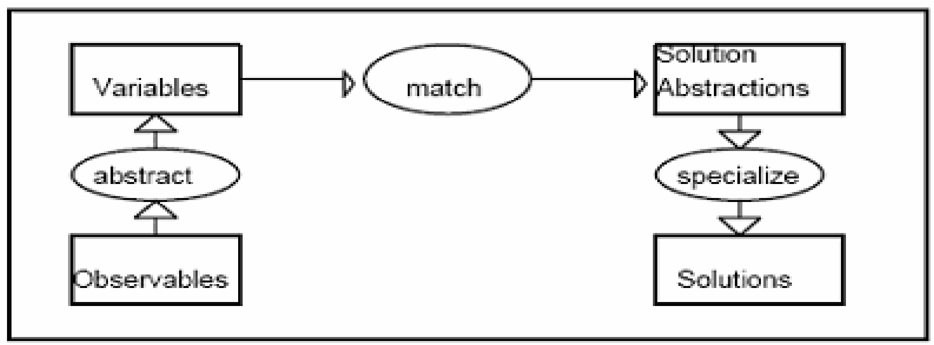


Fig: 2.4 Process flowchart for system development and designing [12]

Knowledge based engineering (KBE) is a method for building product models containing experience of engineering design and evaluation in terms of rules coupled to geometric parameters. Stoke defines KBE as: "The use of advanced software techniques to capture and re-use product and process knowledge in an integrated way."

KBE product models are often built up of objects in an object-oriented programming (OOP) hierarchy of classes and instances. When one object changes in OOP demand-driven capabilities calculate only the dependent objects instead of all parameters, as for procedural programming. KBE is suitable for products where the engineering content is more important than the industrial design content, as the objective is to capture engineering knowledge regarding repetitive and mundane activities. The smaller the difference is between product variants the greater is the reuse of engineering design experience by means of KBE systems. Chapman presents a tool for evaluation of the car structural body in which mesh creation is automated. The Parametric Composite Knowledge System (PACKS) is a tool for composite design that comprises embedded generic manufacturing processes for aerospace structure design. Although both CBR and ES can be coupled to a geometry engine, it is more of a rule that KBE is coupled to a geometric engine. In contrast to ES, KBE focuses on routine, mundane and time-demanding knowledge instead of expert knowledge. KBE enables automatic creation of product definition in terms of geometry and manufacturing plans, for example. It is also possible to enable topological changes; a rectangular prism becomes a cylinder for instance, which is cumbersome if not impossible with traditional parametric solid modelling. Therefore, KBE includes both product synthesis and analysis. KBE systems can generate an infinite number of solutions, as the product model is not dependent on a finite number of cases. As all knowledge is represented as rules in the model, the output will always follow the real process, given that the real process has not changed since it was modelled.

2.6. Knowledge Enabled Engineering Approach [10]

Knowledge Enabled Engineering (KEE) incorporates KBE and other knowledge rich strategies, where the KBE system normally works as the control center due to its demand driven ability. Conceptually, KEE can be explained in three steps:

- capturing of knowledge,
- Automatization and,
- Quality control.

2.6.1. Capturing of Knowledge

In the area of Engineering Design, different kinds of knowledge assets embedded in the engineering design process exist. Different methods need to be used depending on the situation. The first step in the KEE approach is to map the process to be captured, which can then be used to choose what methods and systems are needed to capture it. When the best methods/systems that can capture the process have been identified, the process to follow begins.

2.6.2 Automation

In the end the idea is to automate the process captured. As with the first step different methods/systems exist here as well. By using KEE, the tools needed for each activity are identified. This phase is usually done simultaneously with the capturing of knowledge in the first step. It is an iterative process between the KEE system developer and the end user where the captured process can be shown and discussed.

2.6.3. Quality control

If the process is captured in a computerized system, it can be exactly repeated each time. Several hundred concepts can then be created where the process to generate and evaluate them is the same. This quality assurance gives the engineers a reliable basis to compare concepts. A captured process is now an asset of the company and can be reused whenever needed.

2.7. Methods for KBE application development [27]

Stokes et al. have recently presented the MOKA methodology for KBE application development where, although a total application development cycle is outlined, knowledge acquisition and knowledge formalization are in focus. This method is suitable for larger projects involving many people, since such project entail high regulation, special forms to fill in and require a special knowledge modelling language. Methods have, however, been proposed for projects in smaller organizations. Some claim that KBE application development does not dictate a particular approach, but that many follow a pattern similar to that which is used in normal design engineering problems.

The approach used by (Chapman, C.B. and M. Pinfold) [9] is the following:

1. State problem
2. Identify required information
3. Create key product objects, examples for an automotive structure could be: style, packaging, structural members, joints and panels
4. Define initial conceptual product model, blueprint of KBE application.
5. Build subset of the overall system

The subset of the overall system can than be tested, used and extended with more accompanying subsets in an iterative process that is referred to as rapid application development.

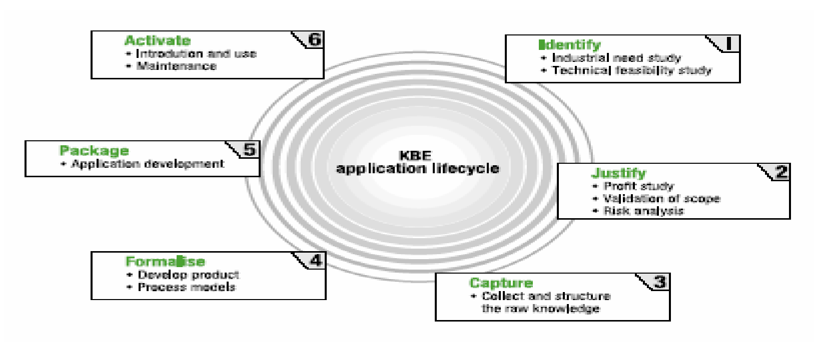


Fig: 2.5. Viscous circle of KBE [1]

2.7.1. Knowledge enabled engineering

Knowledge enabled engineering (KEE) is another approach that can be used to support product development by means of support methods and tools, i.e., KEE applications. KEE includes ED, KBE and similar knowledge-intensive methods and the key concepts are defined as:

The key concepts are that the logics of the design object (artefact) and the actual design process are described in a way that allows generation of design solutions (i.e., geometries and more).

The main difference between KEE and the approach described above is that KEE includes a wider set of knowledge techniques and an engineering design focus. The KEE application development strategy is therefore not restricted to a certain development methods. Instead, the most suitable method is chosen for instance, when developing a large business KBE application, MOKA is suitable. The application development strategy used in this thesis is quite similar to above and is described by presenting how it was applied for development of the tools in this thesis in terms of four steps: support tool need-finding, knowledge acquisition, knowledge formalization and tool implementation.

2.7.2. Support tool need-finding

This activity aims to find the engineering design need, for which a support tool can be developed, i.e., state the problem. This was done through meetings with involved people such as company staff and researchers.

2.7.3. Knowledge acquisition

Knowledge acquisition was performed at the partner company through formal and informal interviews and company reports. Managers and engineers with design, manufacturing, performance and maintenance functions were interviewed and are considered as representatives for the process.

2.7.4. Knowledge formalization

The acquired knowledge was interpreted into rules and formalized to a computer implementation-friendly format. Formalization was done using a partner-company approach. During this activity the implementation structure, i.e., a hierarchical class structure, takes shape.

2.7.5. Tool implementation

When the acquired knowledge was formalized, implementation of the tool in a computer environment was started. Choice of software was adapted to the environment in which the tool should be used. To simplify tool introduction to a partner industry, using partner-industry software is preferable. The KBE module of Unigraphics Solution NX1/NX3, Microsoft Access database and Microsoft Excel were used for flange design tool implementation.

2.8. An overview of knowledge engineering and its relevance to CAAD [17]

Computer-aided architectural design (CAAD) has come to mean a number of often disparate activities. These can be placed into one of two categories: using the computer as a drafting and, to a lesser extent, modeling system; and using it as a design medium.

Architecture is concerned with much more than numerical description of buildings. It is concerned with concepts, ideas, judgment and experience. All these appear to be outside the realm of traditional computing. Yet architects discoursing use models of buildings largely unrelated to either numerical descriptions or procedural representations. They make use of knowledge -about objects, events and processes - and make nonprocedural (declarative) statements that can only be described symbolically. The limits of traditional computing are the limits of traditional computer-aided design systems, namely, that it is unable directly to represent and manipulate declarative, on-algorithmic, knowledge or to perform symbolic reasoning.

Developments in artificial intelligence have opened up ways of increasing the applicability of computers by acquiring and representing knowledge incomputable forms. These approaches supplement rather than supplant existing uses of computers. They begin to allow the explicit representations of human knowledge. The remainder of this chapter provides a brief introduction to this field and describes, through applications, its relevance to computer-aided architectural design.

In the present work an expert system is proposed for the Design and automation of transformer. In relevance to that, there are some brief details about the Expert system.

2.9. Expert system [18]

Expert systems, which are part of knowledge engineering, have been defined as interactive computer programs which use symbolic inferential reasoning to deal with problems that are difficult enough to require significant human expertise for their solution. Thus, expert systems aim to capture the ability of rational human experts to ask pertinent questions, to explain why they are asking these questions and to defend their conclusions and recommendations. These characteristics are unrelated to a specific domain of knowledge and apply to all expert systems. All expert systems share a common fundamental structure even if their knowledge encoding mechanisms differ. Each has the following components:

(1) An inference engine;
(2) A knowledge base;
(3) An explanation facility;
(4) A state description;
(5) (Possibly) a natural language interface; and
(6) (Possibly) a knowledge acquisition facility.

The inference engine provides the mechanisms for driving the system. Many shells use a knowledge representation based on production systems or production rules (Buchanan and Shortliffe, 1985). Prolog is a good vehicle for manipulating production systems. Clearly, more complex controllers are needed to allow data-driven (or forward- chaining) and goal-driven (or backward-chaining) enquiries to be processed. A number of expert-system shells oriented towards computer aided design have been written in Prolog (Gero, 1985) for rule-based knowledge systems.Combined synthesis-evaluation systems can be generated with an appropriate knowledge base. Rivka Oxman[10] in the Computer Applications Research Unit is constructing a system which is the beginning of a designer's apprentice and its example domain is the design of kitchens. It aims to allow the user to construct a design graphically and have the expert system check the design as it proceeds.

2.10. KF Coding and Applications in UG

2.10.1. Continuously Advancing Technology

To remain at par with the cut throat competition one should be at par with the technology as well as with the Data Base. The graph below gives a vague idea about the technology flow in the last 30 years.

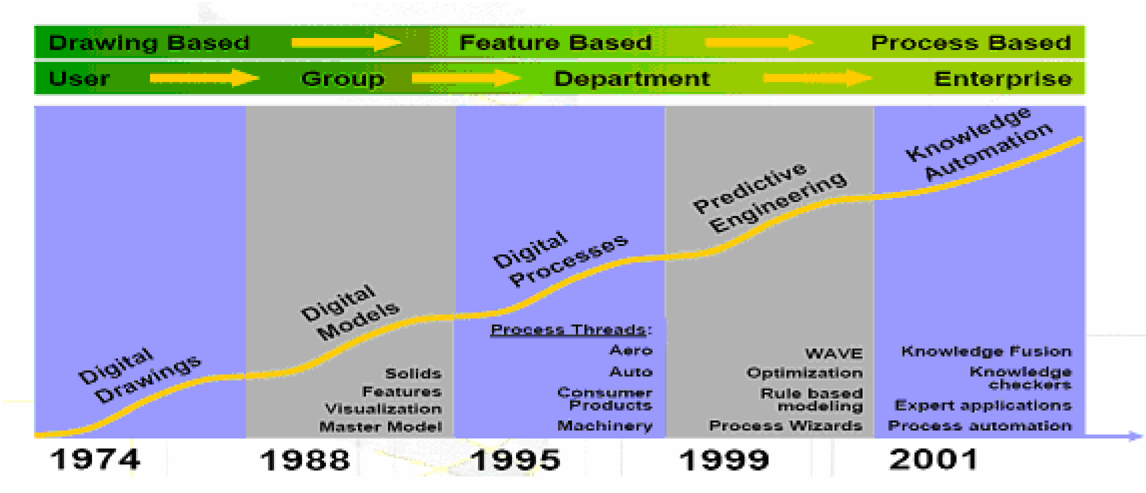


Fig: 2.6. Technology flow in (CAD/CAM &CAE) in last 30 Years [42]

From drawing base in 1974, we have reached at process based engineering in 2001 and to fulfill the requirements of advanced technologies, we have to update the tools, as well as processes of working. So considering the feature industry the research is undergoing on intent product design, where we have to develop the intelligent systems which have the capability not only to execute the commands based on passed experience but also capable enough to take the decision in a new persisting condition. This could be achieved only with the help of Intelligent Systems.

In the present case also, the main motive is to design an Intellegent system which could completely automate the "Design and Automation of a Transformer".

2.11. Intent Product Design: [17]

When one thinks of increasing the production accuracy and efficiency than in the present scenario he has to be compatible with "Intent Product Design". It basically includes:

1. Knowledge Base Design (Knowledge Acquisition and representation)
2. Case Based design (Case Reasoning)
3. Intelligent Conceptual Design
4. Knowledge Mining

1. Knowledge Based Design [43]

There are actually two design problems: "Design-in-the-Small" (DITS) and "Design-in-the-Large" (DITL). Design-in-the- Small describes the activities of a single engineer or small group who select components, determine parameters, perform analyses, etc. In reality, this represents only a fraction of the overall design task.The major issue in "Design-in-the-Large" is how to achieve high levels of collaboration among engineers. That is, how each engineer's design task can be managed so that it integrates well with the results of others.

We are approaching the DITL problem by using knowledge-based information technologies to enhance the degree of awareness, understanding, cooperation, and coordination among engineering team members.For this Knowledge base design the tool we are deciding to use here is Knowledge Fusion in UG and that too using a supported independent customized library using Ufunc and API.

2. Case Based Design [45]

In recent years, researchers in Artificial Intelligence (AI) have studied if *cases* (information about specific problem-solving experiences) could be used as a representation of experiential knowledge. Cases are valid at a specific situation in contrast to generalized knowledge, e.g. rules. Making use of past experience in the form of cases is commonly known as Case-Based Reasoning (CBR) (Kolodner, 1993). The application of CBR in design, known as Case-Based Design (CBD), is still in its infancy even though several CBD-systems focusing on various domains have been developed.

3. Intelligent Conceptual Design [12]

Intelligent design is presented as an alternative to purely naturalistic explanations for evolution. The stated purpose is to investigate whether or not existing empirical evidence implies that life on Earth must have been designed by an intelligent agent or agents. William Dembski, one of intelligent design's leading proponents, has stated that the fundamental claim of intelligent design is that "there are natural systems that cannot be adequately explained in terms of undirected natural forces and that exhibit features which in any other circumstance we would attribute to intelligence.

4. Knowledge Mining [44]

Since a textual format is a very flexible way to describe and store various types of information, large amounts of information are stored and distributed as text. Moreover, the amount of accessible textual data has been increasing rapidly. Such data may potentially contain a great wealth of knowledge. However, analyzing huge amounts of textual data requires a tremendous amount of work in reading all of the text and organizing the content. Thus, the increase in accessible textual data has caused an information flood in spite of hope of becoming knowledgeable about various topics

In order to overcome this counterintuitive situation, a text mining technology called TAKMI (Text Analysis and Knowledge Mining) has been developed to acquire useful knowledge from large amounts of textual data such as internal reports, various technical documents, messages from various individuals, and so on. The most important issue for this text mining technology is how to represent the contents of textual data in order to apply statistical analysis.

So, after making a comparison here I decided to use Knowledge Based design as a tool for achieving the complete automation and design of transformer. This could be achieved in Uni-Graphics using Knowledge Fusion (UG/Open and Ufunc).

The figure below shows the branching of UG/Open (basically a module) that is mainly used for Knowledge Based Customization in Unigraphics. UG/Open basically contains GRIP, UI Styler, API and Menu Script.

2.12. UG/Open API [39]

"Open architecture" is software design that allows access to program functionality without requiring access to the source code.

UG/Open API is a toolkit of C language callable routines for application programmers to write programs that work directly with Unigraphics. Unigraphics solutions provide a library of C language functions; include files and FORTRAN Subroutines that interface with Unigraphics object database (Unigraphics Object Model).

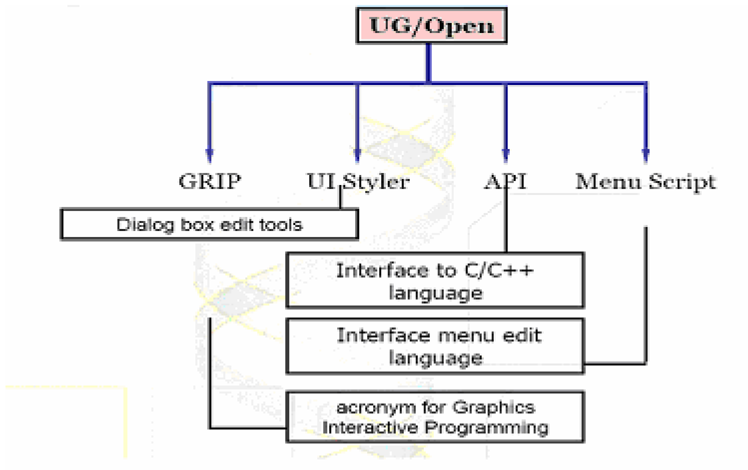


Fig: 2.7. UG/open Tree used for Knowledge based Customization [39]

As shown above that using API ,we can customize the library of UG as per our convenience, by making the required programs in C/VC++ and then converting them to DLL with the help of COM/DCOM i.e. Component Object Modeling or Distributed Component Object Modeling a facility provided in API.

2.12.1. Forms of UG/Open API

UG/Open API programs can be developed in two forms Internal/External.

1. The Internal UG/Open API programme is a relocatable image that runs within the unigraphics interactive process .Programs can be developed to create and access geometry, analyze geometry, create and edit features, create and edit expressions and manage data. Routines to interact with users using standard unigraphics Motif widgets are provided. Access to general motif calls is supported with in the UG/Open API.

2. An External UG Open API application is a program that runs without displaying a graphics window. External UG/Open API application program tend to be more for data management than for geometry manipulation. External UG/Open API application programs can construct and edit models, where no model display is necessary.

So, to summaries, UG open architecture programs can be written as “External” or “Internal” programs. External programs run independently from the UG executable. That is, it is not necessary to start UG to run External programs. External programs use UG code simply for the numerous libraries. Since UG is not running, no graphical interaction is possible unless it is specially programmed in. On the other hand, internal programs are compiled as dynamically linked libraries and must have a function that matches the prototype:

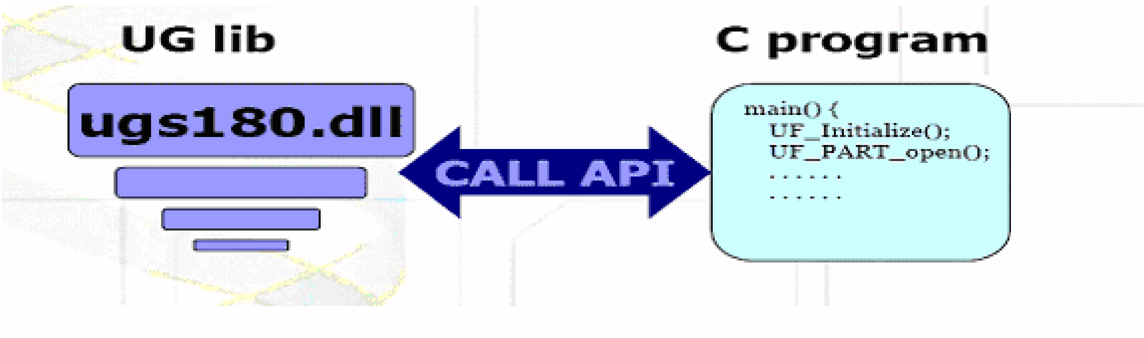


Fig: 2.8. Interfacing of C with UG library using API [39]

The above figure gives a clear view that how (Application Programmer Interface) API is basically used to convert /linking a basic c code for user interface library. Using this we create our Database as well as our library which could be used further during assembly and modeling for performing the different tasks.

2.12.2. UG/Open API Applications

Some useful applications of UG/Open API includes

- Geometry/ Feature creation and editing
- Expression Creation/Manipulation
- Geometry Analysis
- Part standardization
- File management
- Data Access
- Family of parts
- Create and interact with UI styler Dialogs.
- Create/maintain user defined and smart objects.

2.13. GRIP (Graphics Interactive Programming) [13]

GRIP is used to create FORTRAN like programs to operate the Unigarphics systems. Many operations that can be performed interactively can also be performed by using the GRIP program. Commands are available to create, geometric and drafting entities, control preferences, perform file management functions and modify existing

geometries. GRIP also provides the interactive commands. These commands display messages in a motif dialog, allow the user to interact with a GRIP program while it is running. The message displayed by these commands can be programmed to fit the user's specific needs. There are interactive commands to control entity selection, menu option selection, data entry, text entry and generic point sub function.
API CAM's functionality increases significantly. GRIP NC code is now almost redundant. GRIP NC is still officially required to assign Point-to-Point geometry and unofficially required to fill in the blanks for API CAM's bugs missing parameters and missing UDE's.

Almost all of the operation parameters for milling are in place and Point-to-Point is close behind. GRIP NC will soon be redundant. API is rapidly gaining on GRIP NC as the CAM automation language of choice.GRIP as discussed above is mainly used in UG for developing the machining sequences which could further be optimized but now most of the work have carried out in API. So from above discussion and also in the project GRIP is mainly used for creating just user interfaces instead of dealing with big conceptual programs.

2.13.1. Program Development using GRADE [13]

The development of a GRIP program or a subprogram begins when the source file is created. The source file consists of statements, labels and comments arranged in logical manner to perform various tasks.

The file is created by using an operating system editor, typically after entering GRADE, the GRIP advanced Development Environment. GRADE facilitates editing, compiling, and linking to create a GRIP executable file. The executable file can than be run after entering UNIGRAPHICS or by entering GRIP batch.

GRADE is basically a separate, executable program that allows a variety of GRIP program development functions to be performed from the operating system. Finally we can say that GRADE could enhance the capabilities of GRIP up to a large extent and hence always preferred to use as a subprogram with GRIP.

2.13.2. GRIP Application/ field of Scope [13]

(1) GRIP/UNIGRAPHICS

Special Geometric Functions

Calculation and Analysis

Plotting

Part Standarization

File Management

Data Ascess

Family-of-part-programming

(2) GRIP/NC Machining

Basic Machining

Planar Milling

Surface Milling

Lathe Machining

(3) GRIP/ FINITE ELEMENT MODELLING

Node Creation and editing

Element creation and editing

Mesh Node/ Element creation

Mesh definition by surface

2.14. GRIP Vs API [2]

API is the preferred language to any C programmer. It has more overhead than GRIP but it executes faster and can interface with the world out there through the Internet, HTML files, etc. While the instruction sets are long and tedious, it can be easily customized. Generic functions can be created and easily accessed from any other API program or function. Here's an example of come API code that will set up the object mask for points only:

```
selopts.other_options = 0;
selopts.reserved = NULL;
selopts.num_mask_triples = 1;
selopts.mask_triples = mskdata;
selopts.scope = UF_UI_SEL_SCOPE_WORK_PART;
mskdata [0].object_type = UF_point_type;
mskdata [0].object_subtype= 0;
```

```
mskdata [0].solid_type = 0;
UF_CALL (UF_UI_set_sel_mask (seldata.value.selection,
UF_UI_SEL_MASK_CLEAR_AND_ENABLE_SPECIFIC, 1, mskdata));
```

This is a little more tedious than the single line GRIP command that does the same thing. The C language is very friendly, however, when it comes to sharing code among routines. For example, by storing the above code in a function named "Mask Points ()," we can now execute it from the single line command "Mask_Points()". Parameters can be passed to and from the function by including them in the parentheses but we won't go there now.

The point is that this type of function can be created for all the machining operation types and subtypes and you can call them with a single line command packed with the appropriate parameters. I store these functions in a header file. In this way we can create libraries of common functions and call them with C++ classes. Either way, the transition from GRIP, while difficult at first, can be done without too much pain.

One of the major benefits of API CAM is the ability to access the ON interactively during program execution. This makes it possible to select CAM objects in a similar manner to selecting geometry and drafting objects. This facilitates the use of an API program for testing the integrity of operations prior to output. Some companies require this as part of their QA. Currently the ON cannot be refreshed until exiting the API. This is an inconvenience but according to Tue Doan at GTAC it has been proved and will be fixed.

2.15. UI Styler: [39]

User Interface (UI) Styler is a visual user interface builder that makes it possible to interactively design portable NX-style dialogs. Used both internally by UGS developers and externally by users and third-party developers, UI Styler provides the application module, dialog builder, objects, libraries and documentation necessary to interactively create production ready dialogs.

Using a point-and-click interface, users can develop dialogs for Open applications with a minimum amount of time and effort. No previous experience with Windows, SDK or X/Motif is required. UI Styler manages these complexities to create platform-independent dialogs, with interactive object selection and macro support built in.

The main advantage of having UI styler is that neither we have to do a coding for create those boxes like in VB nor we have to do a programming in Graphics to provide a interactive menu. Some of its advantages could be listed as follows:

- Reduces development time
- Helps in Rapid prototyping
- Enables dialogs to be built as per standards
- Provides compatibility with menu scripts

2.15.1. UI/Styler – Introduction [30]

Button	Action
Ok	Accepts input and terminates the dialog
Apply	Accepts input and continues to display the dialog
Back	Does not accept input and reverts to the previous dialog
Cancel	Does not accept input and cancels the dialog and all previous dialogs

Table: 2.1 – Refresh and callback value for UI/Styler

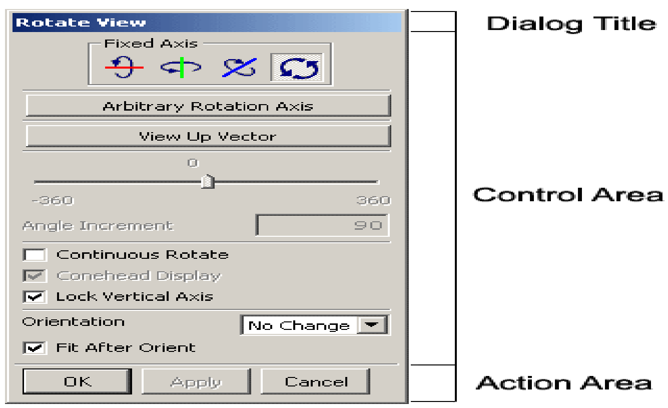


Fig: 2.9. Screen shot of a dialog box [30]

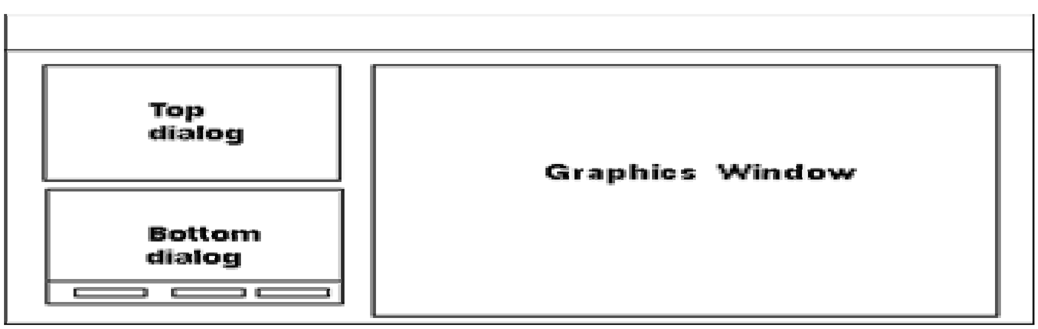


Fig: 2.10. 2D sketch to design a dialog box [30]

(1) Top Dialog

1. Top dialog launches bottom dialogs to handle the gathering and processing of information
2. Acts as an entry point to the application's overall functionality and capabilities
3. Dialogs are normally used as a main dialog to a larger application scope such as Modeling or Drafting

(2) Bottom Dialog

1. Implements the core effort of a particular functionality
2. Provides a larger set of capabilities than a top dialog
3. Should contain navigation buttons and selection capabilities

2.15.2. Steps to Build a Dialog Box:

1. Starting Open User Interface Styler
2. Setting Update Preferences (Optional)
3. Opening a New Dialog (Removes an already existing dialog)
4. Setting Dialog Attributes
5. Setting Dialog Callbacks
6. Setting Dialog Selection Options
7. Inserting a Dialog Item
8. Adding a Bitmap (Optional)
9. Positioning Dialog Items
10. Saving a Dialog
11. Exiting Open User Interface Styler

2.15.3. Styler Data Structure Definition

This code as being developed in GRIP to provide callbacks and refresh to the buttons.

```
typedef struct UF_STYLER_item_value_type_s
{
        int reason;
        const char *item_id;
        int subitem_index;
        int count;
        int item_attr;
        int indicator;
        union
        {
                char *string;
```

```
                char **strings;
                int integer;
                int *integers;
                doublereal;
                UF_UI_selection_p_t *selection;
        } value;
} UF_STYLER_item_value_type_t, *UF_STYLER_item_value_type_p_t;
```

Callback Function Definition –

```
typedef int (*UF_STYLER_cb_f_t) (
            int dialog_id,
            void *client_data,
            UF_STYLER_item_value_type_p_t call_data
                                        );
```

Callback Return Values –

UF_UI_CB_CONTINUE_DIALOG

UF_UI_CB_EXIT_DIALOG

Setting Dialog Values –

```
UF_STYLER_set_value ( int dialog_id,
                      UF_STYLER_item_value_type_t * data )
UF_STYLER_item_value_type_t data;
data.item_id = DEMO_REAL;
data.item_attr = UF_STYLER_VALUE;
data.value.real = 6.90;
error = UF_STYLER_set_value ( dialog_id, &data );
```

The code shown above is developed mainly to set the callbacks, and to set dialog values.

2.16. Concluding remark:

With this, the chapter could be concluded as how KBE has been used and usage of Knowledge Fusion and the languages used in kF like GRIP and API.

Only this helps in providing a viable solution to the existing problem, i.e. in developing an Expert System in true sense. Here Knowledge Fusion with Excel sheets work as a database to store the previous information and default values. API helps in developing subroutines to opt the desired value for the system in the time of need or can say that act as a CPU which provides the logics to take the decision.

GRIP with GRADE provides intended actions and subroutines that help in location the object in the proper class and recognize the component when used in future.

UI/Styler is one of the most useful tools which helps in making the application not only interactive but also provides a pathway for the user to interact with the software in a friendly manner.

Knowledge engineering provides novel tools in the domain of computer-aided architectural design which supplement rather than supplant our existing tools. However, these tools are opening up avenues and approaches to computer-aided design we were previously unaware of or unable to tackle. Issues central to automated and semi-automated synthesis can now be addressed and computationally tractable systems produced. Whilst knowledge-based systems are certainly not a universal panacea they provide new directions for computer-aided architectural design which expands the role of the computer in design.

CHAPTER 3
GROUP TECHNOLOGY
(LITERATURE REVIEW)

Group Technology is a manufacturing philosophy that exploits similarities in the design, fabrication, and assembly attributes of products. A Group Technology (GT) code is an alphanumeric string which represents critical information about the product in a concise manner. Comparing the GT codes of two products is a quick and efficient method for estimating product similarity in selected attributes. GT codes can be used to search a database of products and retrieve the designs and process plans of those products which are similar to a given design, to generate new process plans automatically using a knowledge-based system, and to assess manufacturability of a product design [40].

Find a universal definition for Group Technology (GT) is not an easy task since many have been introduced by a number of people who have written about it. However, the following definition that is:-

"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 time and effort" [40]

S.P. Mitrofanov [1958] [46] formalized that thousands of items produced annually each has different size, shape, material and function, yet there are similarities between components. So, an attempt should be made to classify components into families.

The objectives of Group Technology are best achieved in business concerned with small to medium batch production; these represent a major part of manufacturing industry. The traditional approach to this type of manufacture is to make, use of a functional layout in the factory, i.e. similar machines are grouped according to type.

As a result, a different concept of manufacturing organization and layout has been developed to overcome the difficulties. This is the Group Technology concept whose emphasis lies in reducing the dimension of the situation to be controlled. Instead of being functionally laid out, the factory is divided into smaller cells in such a way that each cell is equipped with all the machines and equipment needed to complete a particular family of components. It has been found that by switching to this type of cellular manufacture, many benefits of flow line production can be attained in a batch production system. The general achievements of Group Technology have been formulated by Thornley and are illustrated in Figure 1. The application of GT to a traditional manufacturing system can usually result in a simpler material flow system, so that a higher transfer rate and easier production planning and control functions can usually be achieved.

3.1. Development of Group Technology [15]

The basic thinking behind Group Technology can be attributed to the Russians, who carried out initial investigations during the 1920s. The progress of GT since then and its gradual adoption in other countries has been traced by Grayson. The work was extended during the war years by Mitrofanov to include work pieces produced in small batches. His major publication on Group Technology first appeared in 1959 and was translated into English in 1966. Mitrofanov proposed that it was possible to produce a theoretical composite component which incorporated all the major features of components belonging to a family, and that a machine could be tooled up to produce the composite component, thus providing the set-ups required for each component in the family.

In the early 1960s, Opitz carried out an investigation into work piece statistics, which showed that although firms manufacture a variety of products, the spectrum of them all was remarkably similar. Based on the findings of this investigation, he established a classification system which enabled components to be codified by means of their geometrical similarity. A number of methods for classification and coding were being investigated at approximately the same time.

The advances in GT have been greatly influenced by the existence of a classification system devised by Brisch and Partners. The Brisch system was originally designed to facilitate variety reduction, component standardization and product rationalization. It was later developed to suit GT requirements. There have been many applications of

GT using the Brisch system and the most successful example was probably that of Serck Audco Valves.

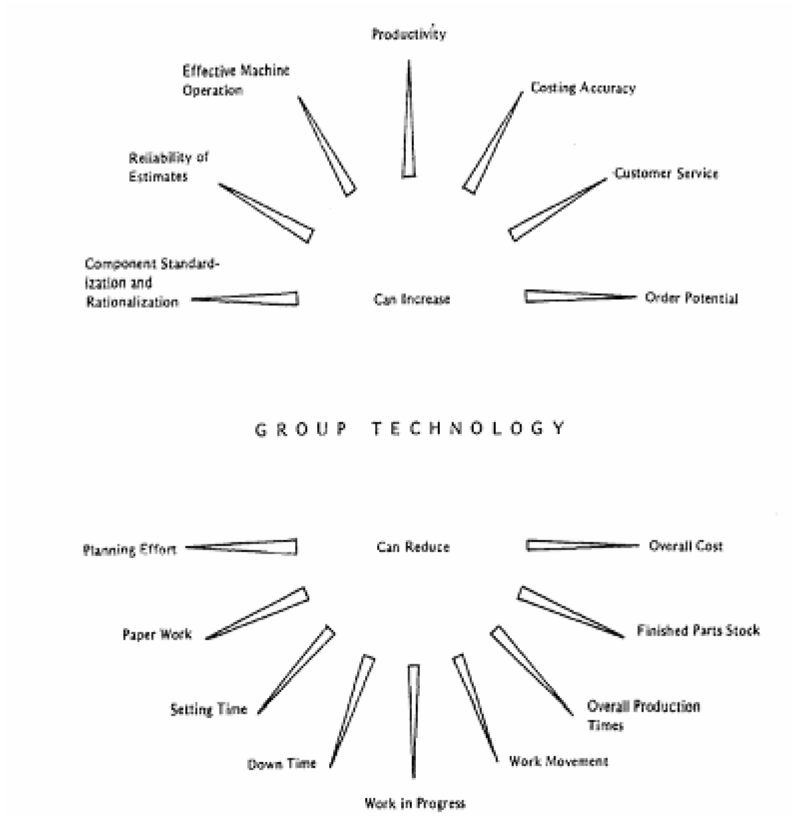


Fig: 3.1. General achievements of Group Technology [15]

3.2. Group Technology Coding and Classification [16]

GT coding and classification schemes attempt to capture design and manufacturing attributes such as the main shape, size, features of the product, production quantity, and material. A large number of GT coding schemes have been developed for discrete machined parts including MICLASS, Opitz and DCLASS [47].

Group Technology codes are typically generated manually or interactively by answering a series of questions and applying appropriate coding rules. However, this is a slow and inconsistent procedure which inhibited the widespread use of Group Technology.

3.2.1. GT coding: [16]

GT is applied by classifying a part into code format. A GT code is a string of alphanumeric characters that represent important design and manufacturing related attributes. A number of GT coding schemes have been developed for mechanical applications but at present, only a few electrical applications have been developed.

The benefits of GT most relevant to agile manufacturing are design retrieval, part classification, design standardization, variant process planning, and design and manufacturability evaluation.

3.2.2. Group Technology in Design: [47]

We know that generating the CAD model of a part design requires specification of its shape (geometry). So it makes sense to categorize, or classify a set of parts based on their shape.

PROBLEM:

How can we decide whether two parts/products are similar?

It turns out that there is no good solution to this problem. However, some classification techniques have been successfully utilized in practice. We shall restrict our attention to individual parts. Similar techniques can (and have) been applied to components/assemblies also.

3.2.3. Coding and Classification Schemes: [47]

(1) ***Coding:*** assignment of a symbol (or a set of symbols) to represent information.

(2) ***Classification:*** a protocol that is used to separate a large group of objects into separate sub-groups.

Therefore, we first estimate (or survey) all possible different shapes we may be dealing with. Next, we identify some "features" that define something distinguishing about the shape of a part.

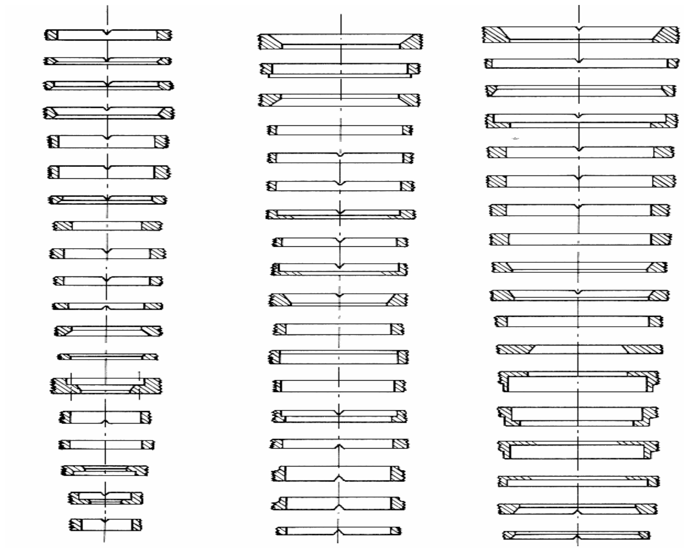


Fig: 3.2. A Design Family [Chang, Wang and Wysk, CAM, pub.: prentice-hall][47]

3.2.4. Examples on different types of coding:

This process is largely intuitive (it would be nice to have mathematical representations of such qualifiers).

Then, for a given application where we wish to apply GT, we select a small set of "representative features". Each of these is given one (or a few) code digits.

For any given feature, we can further classify the variations -- and therefore assign different values to the code digit (say 1, 2, 3, ..., 9) depending on the size of the actual feature.

Based upon this idea, several schemes of coding and classification have been developed:

(a) Hierarchical Structures (Monocodes): [47]

Here, each digit (or position) in the code represents a feature/sub-group. The first digit represents an entire group. The next digit represents sub-groups of the feature, and so on.

In this sense, each subsequent digit is qualified by the preceding digits (or, in an object-oriented sense, each subsequent digit inherits the properties of the previous digits).

Example:

Consider all parts to be classified in terms of a feature: rotational symmetry.

1 == Non-rotational (prismatic) parts

2 == Rotational parts.

Within these groups, we can further classify by feature: presence of hole(s).

0 == No holes

1 == has holes

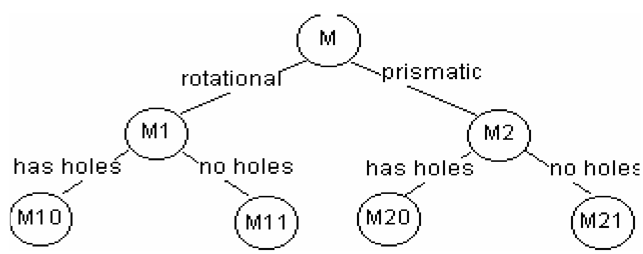


Fig: 3.3. Hierarchical structures for monocodes [47]

Advantages of monocodes:

1. With just a few digits, a very large amount of information can be stored

2. The hierarchical structure allows parts of the code to be used for information at different levels of abstraction.

Disadvantages:

1. Impossible to get a good hierarchical structure for most features/groups

2. Different sub-groups may have different levels of sub-sub-groups, there by leading to blank codes in some positions.

(b) Chain codes (Polycodes): [47]

In this method, the code digit represents one feature. Thus, the value of any given digit (or position) within the code has no relation to the other digits.

Example: A feature to be represented may be the longest linear dimension;

The value of this digit may be represented by an integer that is obtained by rounding off the length (longest dimension) [in inches] of the part being coded. What if we are dealing with parts longer than 9.5 inches?

We may need to reserve two (or more) digits for this feature in that case.

Advantages: 1. Easy to formulate

Disadvantages: 1. less information is stored per digit; therefore to get a meaningful comparison of, say, shape, very long codes will be required.

2. Comparison of coded parts (to check for similarity) requires more work.

(c) Hybrid Structure: [47]

In this case, the code for a part is a mixture of polycodes and monocodes. Such coding methods use monocodes where they can, and use polycodes for the other digits -- in such a way as to obtain a code structure that captures the essential information about a part shape. This is the most commonly used method of coding and classification.

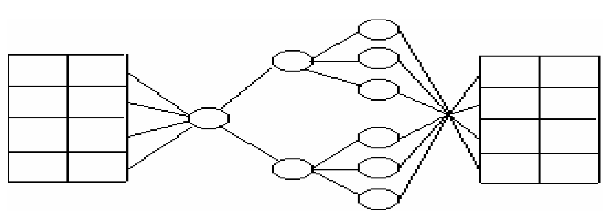


Fig: 3.4. Showing the virtual prototype of Hybrid structures [47]

Two Hybrids coding schemes are: ***Opitz and KK3.***

Properties: 1. Hybrid codes

2. Useful in shape classification for design and machining applications

(d) Opitz Code: [47]

The original Opitz code was a 5-digit code.

Digit/Position	Type	Feature/Group
1	Integer	part class (rotational/non-rotational)
2	Integer	external shape
3	Integer	internal shape
4	Integer	plane surface features and machining
5	Integer	Auxiliary features ()

Table: 3.1 – development of opitz code for bushing

Later, 4 more digits were added to the coding scheme, in order to increase the manufacturing information. These last four digits are also called *supplementary digits*. All four are integers, and respectively represent: [Dimensions, Material, Original shape of raw stock, and Accuracy.]

(e) The KK-3 code: [47]

It was originally developed by the Japan Society for Promotion of Machining Industry. The domain is machining and grinding parts. The code is 21-digits, each of which is an integer.

Digit	Items		(Rotational Components)
1	Parts name		General classification
2			Detailed classification
3	Materials		General classification
4			Detailed 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 primary shape
9			Concentric screw threaded parts
10			Functional cut-off parts
11			Extraordinary shaped parts
12			Forming
13			Cylindrical surfaces
14		Internal surface	Internal primary shape
15			Internal curved surface
16			Internal flat / cylindrical surface
17		End surface	
18		Non-concentric holes	Regularly located holes
19			Special holes
20		Non-cutting processes	
21	Accuracy		

Fig: 3.5. Structure of the KK-3 coding system [47]

3.3. Similarity Coefficient-Based Approaches [34]

In *similarity coefficient methods*, the basis is to define a measure of similarity between machines, tools, design features, and so forth and then use it to form part families and machine groups.

3.3.1. Single-Linkage Cluster Analysis *(SLCA)*: [34]

It is a hierarchical machine grouping method known as *single-linkage cluster analysis* using similarity coefficients between machines. The procedure is to construct a tree called a *dendrogram*.

The similarity coefficient between two machines is defined as the ratio of the number of parts visiting both machines and the number of parts visiting one of the two machines:

$$S_{ij} = \frac{\sum_{k=1}^{N} X_{ijk}}{\sum_{k=1}^{N} (Y_{ik} + Z_{jk} - X_{ijk})}$$

Where: X_{ijk} = operation on part *k* Performed on both machine *i* and *j*,

Y_{ik} = operation on part *k* performed on machine *i*,

Z_{jk} = operation on part *k* performed on machine *j.*

3.3.2. SLCA ALGORITHMS [34]

It helps in constructing dendrograms: A dendrogram is a pictorial representation of bonds of similarity between machines as measured by the similarity coefficients.

The steps of algorithm are as follows:

Step 1: Compute similarity coefficients for all possible pairs of machines,

Step 2: Select the two most similar machines to form the first machine cell,

Step 3: Lower the similarity level (threshold) and form new machine cells by including all the machines with similarity coefficients not less than the threshold value,

Step 4: Continue step 3 until all machines are grouped into a single cell.

EXAMPLE:

Consider the matrix of 5 machines and 10 components given below.

Components

Machines	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
M1	1	1	1	1	1		1	1	1	1
M2		1	1	1					1	1
M3	1				1	1	1			
M4		1	1	1				1	1	1
M5	1	1	1	1	1	1	1	1		

Table: 3.2 showing M/c and component pairing for SLCA

Step 1: Determine similarity coefficients between all pairs of machines.

$$SC_{12} = \frac{5}{9 + 5 - 5} = 0.556$$

Similarity coefficients of machine pairs

Machine pairs	M1 M2	M1 M3	M1 M4	M1 M5	M2 M3	M2 M4	M2 M5	M3 M4	M3 M5	M4 M5
C	0.55	0.30	0.67	0.70	0.00	0.83	0.30	0.00	0.50	0.40

Table: 3.3 – Similarity coefficient for machine pairing

Step 2: Select machines M2 and M4, having the highest similarity coefficients of 0.83, to form the first cell.

Step 3: The next lower coefficient of similarity is between machines M1 and M5. Use these machines to form the second cell.

Step 4: The next lower coefficient of similarity is now 0.67 between machines M1 and M4. At this threshold value machines M1, M2, M4, and M5 will form one machine group. The other possible groups will be evaluated by the same way.

3.4. Pattern recognition: [16]

Another approach to grouping parts into families, or identifying similar parts is pattern recognition [48]. In these studies, data concerning the part geometry are extracted from the CAD database in the form of a file. Using pattern recognition techniques, a processor recognizes the part geometry and determines the part in a GT type code. A similar procedure is adopted for the coding of technological parameters. Having these codes, a procedure computes the distance between two parts. It is proposed to use pattern recognition as a base tool for pattern classification.

The main features and advantage of the system is that it could be used in Research and development of a product even in preliminary phase and by tracing out the similar part in the database the lead times can be cut short exponentially. Using pattern recognition is a prospective method which could be of great use to the organization in the long run. The thing is that as the teach files will goes on increasing ,in the similar manner the capability of the software to recognize the newly ordered part, from the database will goes on increasing .

3.5. Software implementation for Classification of Bushing: [8]

As we have discussed in chapter 4 that GT coding can be applied to classify the bushings, on the basis of creating some part families. From above survey we find that as per the requirements of our application opitz code is considered to be most suitable for visual inspection. In this case user has to answer some of the question and based on opitz code a part family would be generated for the new part. Now using SCLA we would find the similarity index of this opitz code generated with previously stored. Than arrange all the SI in descending order i.e. the part family with maximum SI value will be at the top while other will be under it. This could be achieved using Bubble sorting.

Now there will be a screen in front of user which would ask him to write the required SI level. After getting the input from user the software will search that whether this SI level exists if yes? Than part will be transferred to that part family else a new part family will be created for that part.

The advantage of this can be justified in the terms: that if the part is showing a SI of 88%, that refers to that there is a part in the database which could be modified to the desired one just by making 12% changes. So, there is no need to make a new part ,this will reduces the lead time up to great extent and help in increasing the production rate and reducing the waste.

3.5.1. Application of Opitz code:

As per the model, a table is formulated to develop a 5 digit opitz code for bushing. This code will be stored as a part family in the data base for that particular part. Once a database has been created, then based on same parameters questions will be asked from the user to generate a new opitz code for the new part, and if the new opitz code shows similarity with any of the previously stored part, than that part will be opened on the screen. User can modify this part to achieve the desired model.

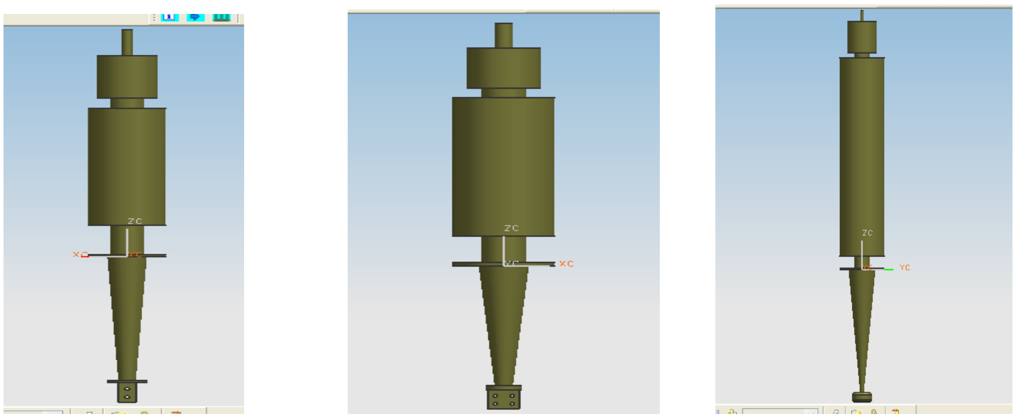


Fig: 3.6. Base plate and other parameters for bushing classification [App-UG]

Bushing:

Length= 4900 Diameter = 2.5” Base Plate= Rectangular, Square, Circular.

5.5.2. Opitz Code Generation for the New Part: [49]

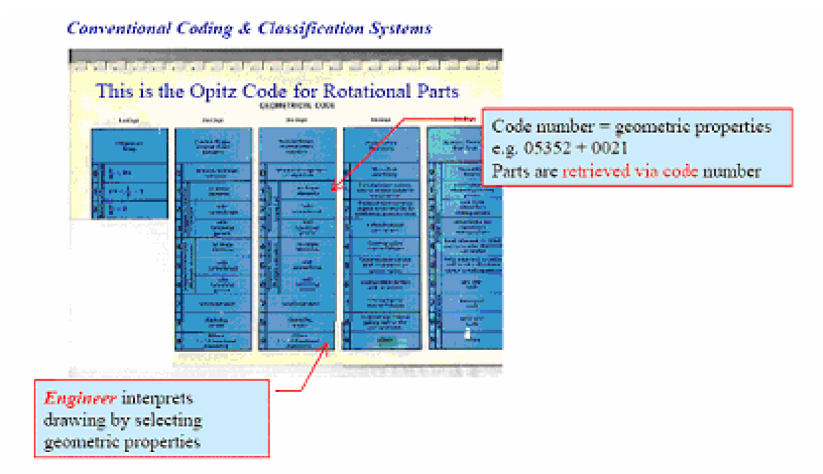


Fig: 3.7. optiz code classification in screen shot from application [49]

3.5.3 .Result from software:

Find Existing part from a sketch:

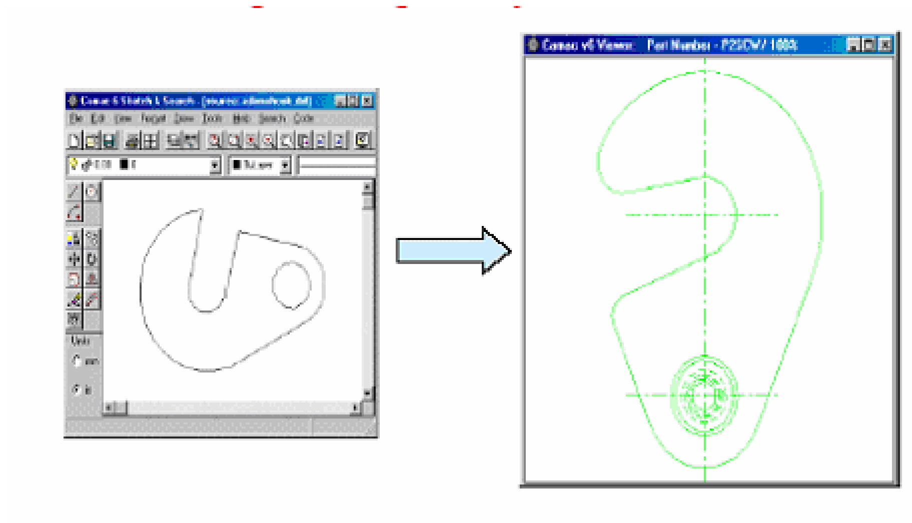


Fig: 3.8. Part from database which could be modified for desired one

3.6. Result and discussion:

Though the results coming out are satisfactory and relevant. It has been also tested that to model 50 bushing in an ordinary manner with logical relations will take 32 hours while if the same work is carried out using the software than it takes only 16 hours .That is an absolute growth of 50%.

But the problem persisting is this that every feature of the part has to be classify by the user , than only a opitz code would be generated. For a small part this would work but for large components like rocket parts ect, the method wastes a lot of time in describing the features and even after that if by mistake a wrong attribute is given than whole results collapsed. So a method is required which could automatically extract the features from the part or scan the image in binary format which could further be used for making the comparison.

The results shows that in which manner user has to provide the information to the software to get the desired results .Though output is relevant but the process become much tedious once the features in the part goes on increasing.

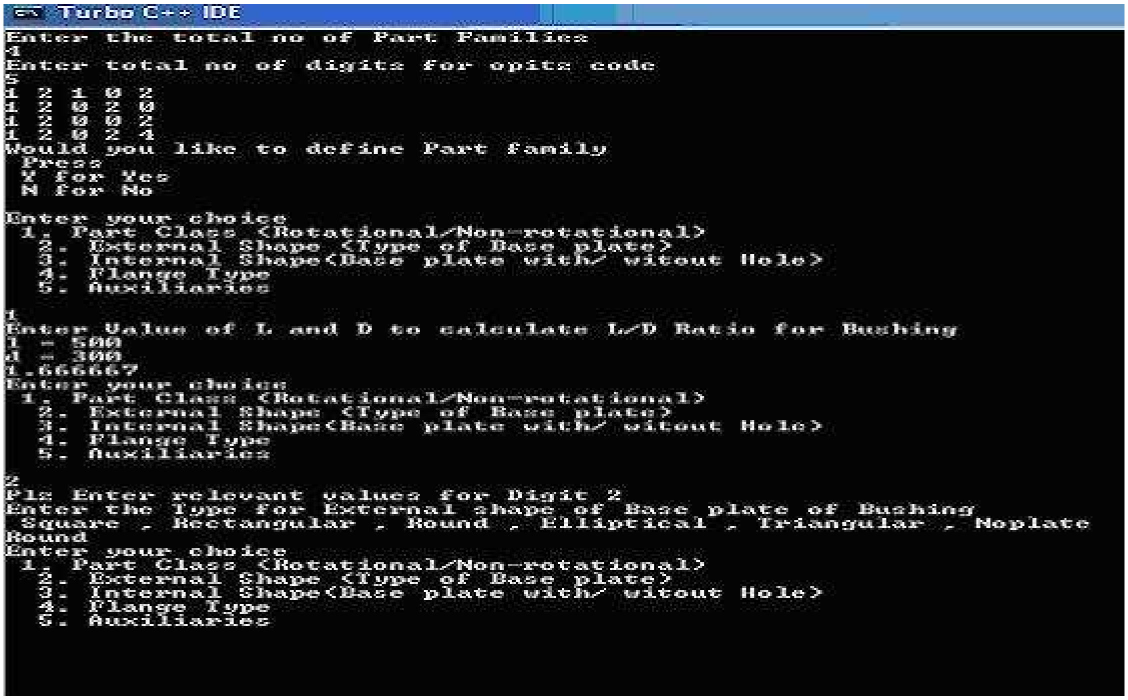


Fig: 3.9. Input screen for visual inspection

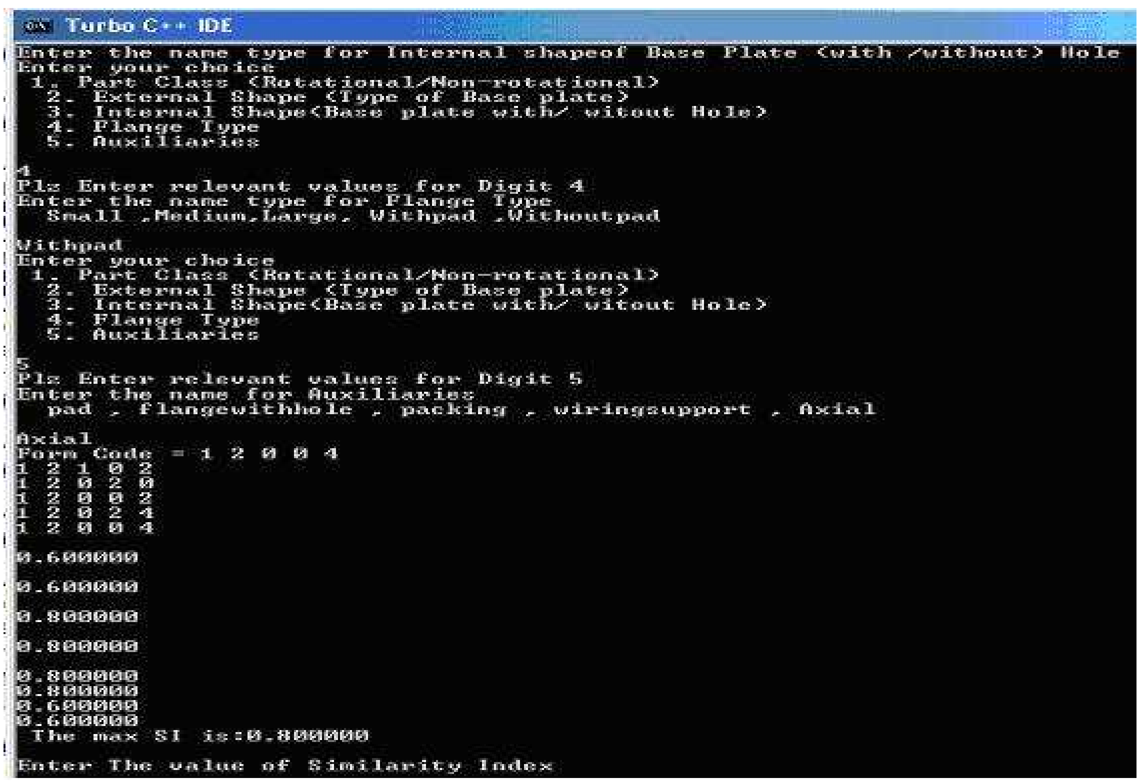


Fig: 3.10. Optiz code developed for new part and it’s SI with the existing one

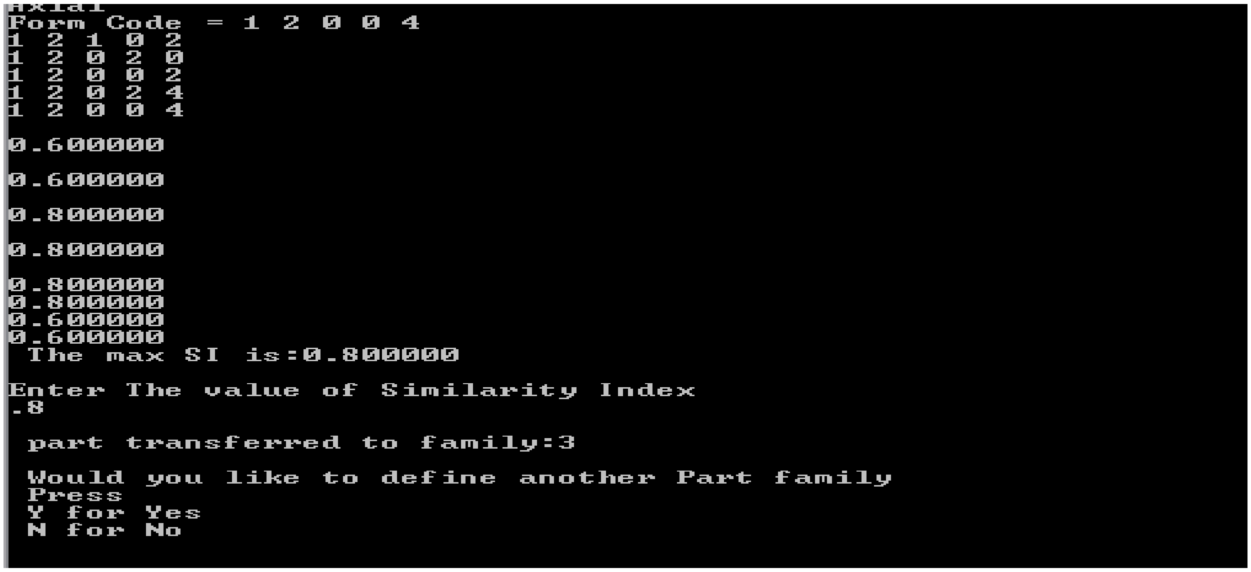


Fig: 3.11. Part family formed and demanding SI value from user

3.7. Gap in above literature and software application:

Above findings shows that group technology is a very useful technique which could be applied for grouping the similar machines, parts, features on the basis of their attributes. On the basis of similarities these can be grouped together, thus saving time and effort. Here as we have applied GT for part classification, because on observation we find that on the basis of similarity in modeling these 78 bushings can be clubbed into 10 models only.

This not only reduces the time and effort but also enhances the overall working process. Secondly the part recognition feature using opitz coding also enhances the search capabilities i.e. if any new part is also coming in the library than that profile could be matched with the previous one and solution could be sorted out in very less time. But to apply the above visual classification of features to 105 parts in not only boring, but even not reliable. This would be a hack-tic job if one has to assemble or classify let say 500 parts .So to resolve the issue an approach of Pattern recognition using AHP will be discussed in next phase of the thesis. In the next chapter we will discuss AHP and than its application in pattern recognition.

CHAPTER 4

ANALYTICAL HIRERACHY PROCESS

(LITERTURE REVIEW)

4.1. Introduction:

Pattern Recognition is one of the main tasks of biological information processing systems, and a major challenge for the coming up technologies in the field of manufacturing like Group Technology, Agile Manufacturing etc. The problem of pattern recognition is to classify objects into categories, given that objects in a particular category may vary widely, while objects in different categories may be very similar. A typical example is handwritten digit recognition. Characters, typically represented as fixed-size images (say 16 by 16 pixels), and must be classified into one of 10 categories using a classification function. Building such a classification function is a major technological challenge, as variability among objects of the same class must be eliminated, while differences between objects of different classes must be identified. Since classification functions for most real pattern recognition tasks are too complicated to be synthesized by hand". Automatic techniques must be devised to construct the classification function from a set of labeled examples (the training set). These techniques can be divided into two camps, according to the number of parameters they are requiring: The Decision Making Techniques which could classify the object on the basis of certain parameters, which could further be ranked as per the ratio of Consistency Index (CI). The memory based" algorithms, which use a (compact) representation of the training set, and the learning" techniques, which require adjustments of a comparatively small but complex and large number of parameters (during training) to compute the classification function.

4.2. Multi Criteria Decision Making (MCDM): [53]

Decision makers are typically requires considering multiple, often conflicting objectives in making decisions. MCDM models are suitable for handling such decision problems. MCDM consists of two related paradigms: Multiple Attribute Decision Making (MADM) and Multiple Objective Decision Making (MODM).MADM problems are assumed to have a predetermined, limited number of decision alternatives .In MODM unlike MADM; the decision alternatives are not given. Instead the set of decision alternatives are explicitly defined by constraints using multiple objective programming.

There are three generic types of solutions for MCDM problems.

Selection: The selection task involves finding a subset A′ of A (set of alternatives) composed of as small as possible number of alternatives

Sorting: The sorting operation consists of assigning each alternative from A to one of the number of predefined categories.

Ranking: The ranking operation involves establishing a preference ranking on the set of alternatives A. The preference ranking represents a priority list of the alternatives.

A number of different structures have been proposed to structure and solve MCDM problems and different and different computational methods developed for their application. The method I going to used to solve the multiple decision based problem of pattern recognition is Analytical Hierarchy Process (AHP), because firstly, It has all the capabilities required to address the specific consideration that are involved in pattern recognition process. It employs relatively uncomplicated and straight forward method for manipulations. Thirdly a large no of modified weight calculating method provides a better way to stimulate deeply inside the problem for greater accuracy and consistency.

4.3. Different Methods to Solve Decision Making Problems: [19]

A large number of methods have been proposed to solve the multi criteria decision making problem, though only some of them able to deal the problem fairly. A brief introduction for each method is surveyed, so that the most suitable method as per the problem definition can be opted.

(a) Cost-benefit analysis: - Cost-benefit analysis (CBA) is a worldwide used technique in decision making. CBA evaluates the costs and benefits of the alternatives on monetary basis. Recently, attempts have been made to incorporate the environmental impacts within CBA to improve the quality of environmental decision making. Although advances have been made, problems persist in applying CBA to environmental issues, including the monetary valuation of environmental impacts (work from: Computer and Automation Institute, Hungarian Academy of Sciences).

(b) Pros and Cons Analysis: - Pros and cons analysis is a qualitative comparison method in which good things (pros) and bad things (cons) are identified about each alternative. Lists of the pros and cons are compared one to another for each alternative.

(c) Conjunctive and disjunctive methods: - These methods require satisfactory rather than best performance in each criterion. The conjunctive method requires that an alternative must meet a minimal performance threshold for all criteria. The disjunctive method requires that the alternative should exceed the given threshold for at least one criterion.

(d) Lexicographic method: - In the lexicographic method criteria are ranked in the order of their importance. The alternative with the best performance score on the most important criterion is chosen.

(e) MAUT methods: - The basis of MAUT is the use of utility functions. Utility functions can be applied to transform the raw performance values of the alternatives against diverse criteria, both factual (objective, quantitative) and judgmental (subjective, qualitative), to a common, dimensionless scale.

(f) Simple Multi attribute Rating Technique (SMART):- SMART is the simplest form of the MAUT methods. The ranking value x_j of alternative A_j is obtained simply as the weighted algebraic mean of the utility values associated with it, i.e.

$$X_j = \sum_{i=1}^{m} w_i a_{ij} / \sum_{i=1}^{m} w_i, \quad j = 1............n$$

(János Fülöp1977) divided whole function for SMART analysis on the scale of 10. Then the final weights are obtained by normalizing the sum of points to one.

(g) The Analytic Hierarchy Process: - The Analytic Hierarchy Process (AHP) was proposed by Saaty (1980). The basic idea of the approach is to convert subjective assessments of relative importance to a set of overall scores or weights. AHP is one of the more widely applied multiattribute decision making methods.

4.3.1. Comparing AHP with others*:* [19] any complex situation that requires structuring, measurement, and and/or synthesis is a good candidate for AHP. However, AHP is rarely used in isolation. Rather, it is used along with, or in support of other methodologies. For example to synthesize the results of other methodologies such as in deciding how many servers to employ in a queueing situation taking into account waiting times, costs, and human frustrations, or to derive probabilities for a decision tree. Broad areas where AHP has been successfully employed include: selection of one alternative from many; resource allocation; forecasting; total quality management; business process re-engineering; quality function deployment, and the balanced scorecard (Saaty, 1980; Buffalo University).

4.4. Analytical Hierarchy Process: [4]

The AHP, developed by Saaty, is a means of weighting or prioritizing impacts through a systematic representation of a problem. Through pairwise comparisons, the relative importance, or weights, of different factors can be measured; tradeoffs between objectives are explicitly considered in these pairwise comparisons. The pairwise comparison process imposes rigor that is missing when directly assigning weights to a number of impacts, because possible inconsistencies in the judgments can be calculated and reexamined. Even with subjective criteria, the weights obtained through the AHP are "ratio scale numbers and correspond to so-called hard numbers (Saaty and Kearns, 1985,).

One foundation of the AHP is the observation that human decision making is not always consistent. The AHP provides a standard by which consistency can be measured. If inconsistency exceeds an established threshold then participants can reexamine their judgments.

4.4.1. Structuring the Problem: [4]

The first step in the AHP is to decompose the problem into a dominance hierarchy (figure 1). The top-most level represents the goal or focus of the problem. Intermediate levels are the criteria on which lower levels depend, and the lowest level is the list of choices or alternatives (Saaty and Kearns, 1985). As many levels as necessary can be used. The lower levels act as the criteria or factors contributing to the level immediately above (see Saaty and Kearns, 1985 for applications of AHP). In the context of the pattern matching problem, the goal is to recognize a image from the VDU, so that this could be further used for part classification in a industry. The alternatives would a CAD /STEP procedure or Image recognition systems but those are too bulky and slower as compared to one, I m going to discuss.

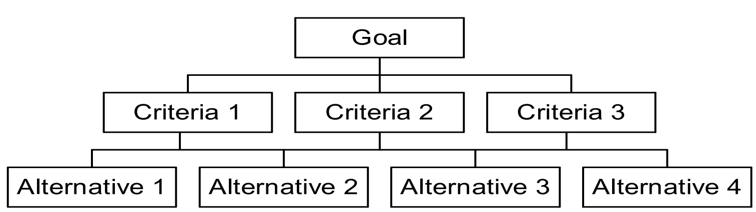


Fig: 4.1.Generic Decomposition of a Problem into a Dominance Hierarchy [4]

Perhaps the most creative task in making a decision is to choose the factors that are important for that decision. In the Analytic Hierarchy Process we arrange these factors, once selected, in a hierarchic structure descending from an overall goal to criteria, sub criteria and alternatives in successive levels.

When constructing hierarchies one must include enough relevant detail to:

(1).Represent the problem as thoroughly as possible, but not so thoroughly as to lose sensitivity to change in elements.

(2) consider the environment surrounding the problem;

(3) identify the issues or attributes that contribute to the solution;

(4) Identify the participants associated with the problem.

Arranging the goals, attributes, issues, and stakeholders in a hierarchy serves two purposes. It provides an overall view of the complex relationships inherent in the situation; and helps the decision maker assess whether the issues in each level are of the same order of magnitude.

4.4.2. Scales of measurement- Avoiding mere number crunching: [35]

One set of numbers pleases a group of people who might be equally pleased with another set of numbers that contradicts the recommendations of the first set. This is mere number crunching. If a decision support theory is to be trustworthy there must be uniqueness in the representation of judgments, the scales derived from these judgments, and the scales synthesized from the derived scales. The Analytic Hierarchy Process is rigorously concerned with the scaling problem and what sort of numbers to use, and how to correctly combine the priorities resulting from them. A scale of measurement consists of three elements: A set of objects, a set of numbers, and a mapping of the objects to the numbers.

A standard scale can be used to measure object or events with respect to the property for which a scale is designed to measure. Since the unit is arbitrary, one can have different numbers to which the objects are mapped. Because a standard scale is not unique, it is important to interpret the meaning of the numbers used in the scale. Thus, in general, the numbers obtained from such a scale are merely stimuli for the memory and have no intrinsic significance. However, most carefully designed standard scales are helpful in that they preserve certain numerical relations in the measurements of the objects, giving us a better way to interpret the stimuli they are measuring than arbitrary scales.

4.4.3. Paired comparisons as ratios: [36]

When we measure something with respect to a property, we usually use some known scale for that purpose. A basic contribution to the subject of this paper, the Analytic Hierarchy Process (AHP) is how to derive relative scales using judgment or data from a standard scale, and how to perform the subsequent arithmetic operation on such scales avoiding useless number crunching.

One of the uses of a hierarchy is that it allows us to focus judgment separately on each of several properties essential for making a sound decision• The most effective way to concentrate judgment is to take a pair of elements and compare them on a single property without concern for other properties or other elements. This is why paired comparisons in combination with the hierarchical structure are so useful in deriving measurement.

The absence of units is another advantage: a comparison is a relative value or a quotient a / b of two quantities a and b of the same kind. The decision maker does not need to give a numerical judgment; instead a relative verbal appreciation, more familiar to our daily live, is sufficient. Saaty has proposed a comparison scale of nine levels (see Table 1)

Intensity of Importance	Definition
1	Equal Importance
2	Weak
3	Moderate Importance
4	Moderate Plus
5	Strong importance
6	Strong Plus
7	Very strong or demonstrated importance
8	Very, very strong
9	Extreme importance

Table: 4.1 Scale of Relative importance [36]

However Saaty's scale sometimes suffers from shortcomings [21, 27]. Some decision problems need a larger scale. AHP also suffers from other criticisms:

(1).The number of required comparisons could be very high,

(2) .Only consistent or near consistent comparison matrices can be used to calculate meaningful priorities which could oftentimes urge a reconsideration of the entries,

(3). the priorities depend on the used method to derive them, (4). Incomparable alternatives are not allowed

4.5. Mathematical Model of AHP: [22]

The first step of AHP is to divide the decision problem into sub-problems, which are structured into hierarchy levels. The number of levels depends on the complexity of the initial problem. The leaves contain the possible alternatives and the inner nodes represent the criteria. To compute the importance of possible choices, pair wise comparison matrices are utilized for each criterion. The element a_{ij} of the comparison matrix A represents the relative importance of choice i, with respect to choice j . Concerning that the element a_{ji} is the reciprocal of a_{ij}.Now the importance value (t) of choice y be expressed as a linear combination of the importance values for each applied criterion:

$$t(y) = \sum_{j=1}^{n} w_j t(y_j) \text{ -- (1)}$$

Where w_j is the importance of choice y with respect to the criterion y_j.Using comparison matrices AHP propagates the value of each node from the topmost (focus) level towards the alternatives, and select the alternative with greatest importance value as its final decision.

4.5.1. Computation of weights: [22]

Let us now focus on the computation of the weights (w) for a selected criterion. The elements of a given pair wise comparison matrix approximate the relative importance of the choices, thus

$$a_{ij} \approx w_i / w_j \text{ -- (2)}$$

Where the elements of unknown vector (w) are importance values. A matrix M is called consistent if its components satisfy the following equalities:

$$d_{ij} = 1/d_{ji} \text{ -- (3)}$$

and $d_{ij} = d_{ik} * d_{kj} \quad \forall 1, j, k$ -- (4)

If A is not consistent, it is not possible to find a vector w that satisfies the equation

$$a_{ij} = w_i / w_j \text{ -- (5)}$$

Now let us define the matrix of weight ratios by:

$$W=\begin{pmatrix} w_1/w_1 & w_1/w_2 & w_1/w_3 & & w_1/w_n \\ w_2/w_1 & w_2/w_2 & w_2/w_3 & & w_2/w_n \\ w_3/w_1 & w_3/w_2 & w_3/w_3 & & w_3/w_n \\ w_n/w_1 & w_n/w_2 & w_n/w_3 & & w_n/w_n \end{pmatrix} \text{------------------------- (6)}$$

or in the matrix notation

$$W = w * w^T \text{ -- (7)}$$

Note that equation (3) and (4) will hold the matrix W:

$$w_{ij} = w_i / w_j = w_i / w_k * w_k / w_j = w_{ik} * w_{kj} \text{------------------------------------ (8)}$$

Hence the matrix of weight ratios is consistent. Because the rows of matrix W are linearly dependent, the rank of the matrix is 1, and there is only one nonzero eigenvalue. Knowing that the trace of a matrix is invariant under similarity transformations, the sum of diagonal elements is equal to the sum of eigenvalues, which implies that the nonzero eigenvalue $\lambda_{\max}$ equals the number of the rows:

$$\lambda_{\max} = n \text{-- (9)}$$

So finally we have the equation for finding the global weights to be solved is:

$$\begin{bmatrix} A_1 \\ A_2 \\ A_3 \\ A_n \end{bmatrix} \begin{pmatrix} w_1/w_1 & w_1/w_2 & \ldots\ldots\ldots\ldots & w_1/w_n \\ w_2/w_1 & w_2/w_2 & \ldots\ldots\ldots\ldots & w_2/w_n \\ w_3/w_1 & w_3/w_2 & \ldots\ldots\ldots\ldots & w_3/w_n \\ w_n/w_1 & w_n/w_2 & \ldots\ldots\ldots\ldots & w_n/w_n \end{pmatrix} . \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_n \end{bmatrix} = n \begin{pmatrix} w_1 \\ w_2 \\ w_3 \\ w_n \end{pmatrix} \text{--------- (10)}$$

The foregoing formulation has the advantage of giving us the solution. But it also gives rise to a far reaching theoretical interpretation. We have multiplied A on the right by the vector of weights {$w = (w_1, w_2 \; . \; . \; . \; . \; . \; w_n)^T$} . The result of this multiplication is *nw.* If n is an eigenvalue of A, then w is the eigenvector associated with it. Now A has rank one because every row is a constant multiple of the first row. Thus all its eigenvalues except one are zero. The sum of the eigenvalues of a matrix is equal to its trace, the sum of the diagonal elements, and in this case, the trace of A is equal to n. Therefore, n is the largest, or principal, eigenvalue of A.

The solution of *Aw = nw,* called the principal right eigenvector of A, consists of positive entries and is unique to within a multiplicative constant. To make w unique, we normalize its entries by dividing by their sum. It is clear that if we are given the comparison matrix A, we can recover the scale. In this case the solution is the normalized version of any column of A.

Now from equation (9) it is straight forward to check that vector (w) is an eigenvector of matrix W corresponding to the maximum Eigenvalue.

$$Ww_i = \sum_{j=1}^{n} W_{ij} w_j \qquad = \sum_{i=1}^{n} w_i / w_j (w_j)$$

$$= \sum_{j=1}^{n} w_i$$

$$= nw_i$$

The aim of AHP is to resolve the weight vector w from a pair wise comparison matrix A, where the elements of A correspond to the measured or estimated weight ratios. Following Saaty we shall assume that

$a_{ij} > 0$, -- (11)

and $a_{ij} = 1/a_{ji}$ --(12)

From matrix theory it is known that a small perturbation of the coefficients implies a small perturbation of the eigenvalues. Hence we still expect to find an eigenvalue close to n, and select the elements of the corresponding eigenvector as weights. It can be proved that $\lambda_{\max}$ > n, and the matrix A is consistent if and only if $\lambda_{\max}$ = n. A way of measuring the consistency of the matrix A is by defining the *consistency index* (CI) as the negative average of the remaining eigenvalues:

$CI = \sum_{\lambda < \lambda_{\max} / n-1} \lambda = \lambda_{\max - n} / n - 1$ -- (13)

4.5.2. Proof and derivation for above consistency check: [36]

We now show the interesting, and perhaps surprising result that inconsistency throughout the matrix can be captured by a single number $\lambda_{\max}$ - n, which measures the deviation of the judgments from the consistent approximation.

Let $a_{ij} = (1+\delta_{ij}) w_i / w_j \quad \delta_{ij} > -1$ be a perturbation of w_i / w_j, where w is the principal eigenvector of A.

Theorem 1. $\lambda_{\max}$ >=n:

Proof: Using $a_{ji} = 1/a_{ji}$, and Aw= $\lambda_{\max}$ w, we have

$$\lambda_{\max} - n = 1/n \ (\sum_{1 \le i < j < n} \delta_{ij}^{\ 2} / (1+\delta_{ij}) \ge 0$$

Theorem 2. *A is consistent if and only if* $\lambda_{\max}$ = n

Proof: If A is consistent, then because of (1), each row of A is a constant multiple of a given row. This implies that the rank of A is one, and all but one of its eigenvalues λ_i , where i=1…n is 0. However, it follows from our earlier argument that: $\sum_{i=1}^{n} \lambda_i = Trace(A) = n$ Therefore $\lambda_{\max}$ = n

Conversely:

$\lambda_{\max}$ = n, implies $\delta_{ij} = 0\, and\, a_{ij} = w_i / w_j$

4.5.3. Physical Interpretation of CI:

For the consistency index (CI), we adopt the value ($\lambda_{\max}$ -n)/ (n-1). It is the negative average of the other roots of the characteristic polynomial of A. This value is compared with the same index obtained as an average over a large number of reciprocal matrices of the same order whose entries are random. If the ratio (called the consistency ratio CR) of CI to that from random matrices is significantly small (carefully specified to be about 10% or less), we accept the estimate of w. Otherwise, we attempt to improve consistency.

4.6. Another Approach to calculate weight using Multiple Attribute Decision Making: [20]

Multiple attribute decision making (MADM) refers to the problem of selecting among alternatives associated with multiple attributes. It is a problem with extensive theoretical and practical backgrounds. The following notations are used to represent a MADM problem:

$S = \{S_1, S_2, S_3, ...S_m\}$: a discrete set of m possible alternatives.

$P = \{P_{1,} P_2, P_3 ....P_n\}$: a set of n attributes. The attributes are objective and additively independent.

$w = (w_{1,} w_{2,} w_3, ...w_n)^T$ The vector with attribute weights

Where $\sum_{j=1}^{n} w_j , w_j \geq 0 \quad \forall j$;

$A = [a_{ij}]_{m*n}$: is decision matrix a_{ij} is a consequence with a numerical value for alternative i, with respect to attribute j, i=1……….m, j=1………..n;

In decision matrix *A*, every a_{ij} is an objective value between 0 and 1. It allows each attribute to have the same range of measurement. This is achieved by normalizing every element in matrix $A=[a_{ij}]_{m*n}$.into a corresponding element in matrix $B=[b_{ij}]_{m*n}$ using the following formulas:

(1) $b_{ij}=(a_{ij}-a_j^{\min})/(a_j^{\max}-a_j^{\min})$ For i=1…..m, j=1……n

For benefit attribute.

(2) $b_{ij}=(a_j^{\max}-a_{ij})/(a_j^{\max}-a_j^{\min})$ i=1………m , j = 1……….n

For cost attribute

(3) $a_j^{\max} = \max\{ a_{1j}, a_{2j}, \ldots\ldots\ldots a_{mj} \}$

(4) $a_j^{\min} = \min\{a_{1j}, a_{2j}, \ldots\ldots\ldots a_{mj}\}$

The decision maker (DM) is to choose *M* (<*m*) most preferred alternatives or the most preferred alternative S^* from the set *S,* $S^* \in S$. Several approaches have been proposed to determine weights .Majority of them can be classified into subjective approaches and objective approaches depending on the information provided.

(a) The subjective approaches select weights based on preference information of attributes given by the DM; they include eigenvector method, weighted least square method, and Delphi method etc.

(b) The objective approaches determine weights based on the objective information (e.g. decision matrix *A*), they include principal element analysis [4], entropy method [11], and multiple objective programming models. But here I am going to drive a integrated method which includes the advantages of both i.e. Objective method and Subjective method. This not only increases the accuracy as well as the efficiency of the system for Decision making.

4.6.1. INTEGRATED APPROACH TO DETERMINE WEIGHTS [20]

Here we are considering Multi attributes decision making method (MADM).Since group classification based on pattern recognition is a multiple decision making problem based on certain parameters. Here the pair wise comparison matrix on attributes is given by a DM, and all the attributes are objective.

In the attribute pair wise comparison matrix $D=[d_{kj}]_{m*n}$, the weights determined by the above subjective approach reflect the subjective consideration of the DM.

In the objective decision matrix $A=[a_{ij}]_{m*n}$, the weights determined by the above objective approach reflect the objective information. In order to make weights reflect both subjective and objective factors, we integrate the subjective approach and the objective approach to set up the following, Two-objective programming model:

Minimize: $\left[\begin{array}{c} Z_1 = w^T F w \\ Z_2 = w^T H w \end{array} \right]$

Subject to $e^T w = 1$ and $w >= 0$

The linear weighted summation method [5] in multiple objective programming analyses is used to solve the problem. The procedure is described as follows:

For integrated method we have to minimize Z_3 as earlier in subjective and objective cases Satty has minimized Z_1 and Z_2

From Z_1 and Z_2 minimization of Z_3 is subject to that for this

$e^T w = 1$ 1.1

For this

$w >= 0$ 1.2

In 1 let $Q = \alpha F + \beta H$ 1.3

Where elements in Matrix Q are:

$q_{ii} = \alpha(n - 2 + \sum_{k=1}^{n} d_{ki}{}^2) + \beta \sum_{k=1}^{m} (b_i^* - b_{ij})^2$ 2

$q_j = -\alpha(d_{ij} + d_{ji})$ 3

Where i, j= 1..............n.

So here α and β denote relative importance of subjective and objective approach respectively and satisfy:

0< α and $\beta < 1$;

Since $\alpha + \beta$ =1;

To solve 1.1, 1.2, 1.3 we have to ignore non-negative constraints of w>=0; and then set up the following Lagrangian function

$L_3 = w^T Q w + 2\lambda_3 (e^T w - 1)$ .... (a)

Where λ_3 is Lagrange's multiplier.

Let $\partial L / \partial w = 0$ and $\partial L / \partial \lambda_3 = 0$ then

$Qw + \lambda_{3e=0}$ 4

$e^T w = 1$ 5

From differential equation solution to equation 4 and 5 could be given as:

$w^* = Q^{-1}e / e^T Q^{-1} e$ 6

$\lambda^*_3 = -1 / e^T Q^{-1} e$ 7

Where w^* is the weight vector determined by the subjective and objective integrated approach.

In general the weight vector determined by equation 7 has no practical significance until it satisfies the constraint w^* >0.

4.6.2. Theorem 1. *Let* $D = [d_{kj}]_{m*n}$ *be a pair wise comparison matrix. If for any i and j, there is at least one* $b^*_j \neq b_{ij}$*, then matrix Q is positive definite and is also invertible or non-singular.*

Proof:-

So first I am going to drive this neglecting the condition that w>=0 , than it could be proved that it is always greater than 0 for a consistent matrix.

From equation 1.3 and 2

$$z_3 = w^T Q w$$

$$= \sum_{j=1}^{n} [\alpha(\sum_{k=1}^{n} d^2{}_{kj} + n - 2) \ + \beta \sum_{i=1}^{m} (b_j^* - b_{\ ij})^2] w^2{}_j \ - \alpha \sum_{k=1}^{n} \ \sum_{j=1}^{n} (d_{kj} \ + d_{jk}) w_k w_j$$

$$= \sum_{j=1}^{n} [\alpha(\sum_{k=1}^{n} d^2{}_{kj} + n) + \beta \sum_{i=1}^{m} (b_j^* - b_{\ ij})^2] w^2{}_j \ - \alpha \sum_{k=1}^{n} \ \sum_{j=1}^{n} (d_{kj} \ + d_{jk}) w_k w_j - \alpha \sum_{j=1}^{n} (d_{ij} + d_{jj})^2 w_j{}^2$$

$$= \sum_{j=1}^{n} [\alpha(\sum_{k=1}^{n} d_{kj}{}^2 + n) + \beta \ \sum_{i=1}^{m} (b_j{}^* - b_{ij})^2] w_j{}^2 - \alpha \sum_{k=1}^{n} \sum_{j=1}^{n} (d_{kj} + d_{jk}) w_k w_j$$

$$= \alpha \sum_{j=1}^{n} (\sum_{k=1}^{n} d_{kj}{}^2 + n) w_j{}^2 \ + \beta \sum_{j=1}^{n} \sum_{i=1}^{m} (b^* - b_{ij})^2 w_j{}^2 \ - 2\alpha \sum_{k=1}^{n} \sum_{j=1}^{n} d_{kj} w_k w_j$$

$$= \alpha \sum_{k=1}^{n} \sum_{j=1}^{n} (d_{kj}{}^2 w_j{}^2 - 2 d_{kj} w_k w_j + w_j{}^2) \quad + \beta \sum_{j=1}^{n} \sum_{i=1}^{m} (b_j{}^* - b_{ij})^2 w_j{}^2$$

Since for any *i* and *j*, there exists at least one $b^*{}_j \neq b_{ij}$, so *z*3 >☐0 holds. The symmetry of matrix *Q* and the definition of positive definite quadratic form indicate that *Q* is positive definite. By the property of positive definite matrix, *Q* is invertible.

So the proof verifies that whatever assumptions we have taken are true and hence the integrated approach would word work with same consistency as subjective and objective approach but with greater accuracy.

4.7. DERIVATION OF THE PRIORITIES [5]

Different methods have been developed to derive priorities. They can be divided into two groups: the eigenvalue approach and the methods minimizing the distance between the user-defined matrix and the nearest consistent matrix.

4.7.1. Eigenvalue approach [5]

Saaty proposes the principal eigenvector **p** as the desired priorities vector. It is derived from the following equation:

$$A.p = \lambda.p$$

Where **A** is the comparison matrix

p is the priorities vector

λ is the maximal eigenvalue

Saaty justifies the eigenvalue approach for slightly inconsistent matrices with the perturbation theory, which says that slight variations in a consistent matrix imply slight variations of the eigenvector and the eigenvalue.

The rank reversal problem for scale inversion is the most pertinent criticism of the eigenvalue method [15]. The solution of the eigen equation (3) gives the right eigenvector **p**, which is not necessary the same as the left eigenvector **p'**, solution of $p'^T * A = \lambda * p^T \Leftrightarrow A^T * p' = \lambda * p'$. The solution depends on the formulation of the problem! This right and left inconsistency (or asymmetry) arises only for inconsistent matrices with a dimension higher than three.

4.7.2. Geometric mean [5]

The priorities are given by the geometric mean. The geometric mean minimizes the logarithmic error [49]:.the formula shown below is mainly used to minimize the error.

$$\sum_{i=1}^{n}\sum_{j=1}^{n} (\ln(a_{ij}) - \ln(w_i / w_j))^2$$

Where a_{ij} : is the comparison between i and j

w_i is the priority of i

n is the dimension of the matrix

Finally this could be used in forming the Rank Matrix which would be a combination of Local and Global priority vectors.

4.8. Concluding Remarks:

An overview of the brief characteristics and details of AHP discussed above can be used to solve a variety of Decision making complex problems.

Framework for applying AHP in the problem, “Application of AHP in pattern recognition (as a classifier) from a Visual display Unit that can be used for part classification in Group Technology (GT)” will be discussed in concurrent chapter.

CHAPTER 5

APPLICATION AND FRAME WORK FOR AHP USING SIMILARITY CONCEPT AND PATTERN RECOGNITION

5.1. Introduction:

In conventional application of Group Technology classification codes, the most broadly applied method is the hybrid method, the combination of hierarchal code and chain code. The hierarchal code is basically a vertical-search device, where as a chain code combines with it, enables horizontal search. The combination of them, known as hybrid structure, is the most commonly used industry coding method.

A special type of chain code, the binary code is receiving more and more attention .Usually in a binary coding system the 0's used for absence of feature and 1's used for processing this feature. Compare with the decimal or hexadecimal coding system it is relatively easy to develop a binary coding system for a special manufacturing environment. Also its digitized binary form is easy to process by a computer .Figure1.1 below shows basic structures of various coding methods.

5.2. Group Technology and Part Family formation: [50]

As we have seen before that the key idea of group technology concept is part family. Properly forming meaningful part families is a critical step for implementation of group technology. [Hem .at.el. 1985] defined GT as:

"A group of parts that has some specific sameness or similarities in design feature or production processes."

Accordingly part family can be generally formed, based on two types of similarities, those of design features like geometric shape, size, material or those of production processes like the process machine tool etc. Basically there are 5 methods currently used for part family classification;

(1) Visual Search

(2) Production flow anlysis
(3) Classification and coding system
(4) Mathematical programming
(5)AI methods

The latest development of part family formation involves the using of Pattern recognition techniques .Han & Ham developed a multi objective clustering method for GT part family formation. In this goal programming based method ,a "diatance function" which checks the similarities or differences among the products was defined .The diatance measurement approach was also applied by some other researchers also .Beside the distance measurement the PR techniques like Entropy analysis (Watanbe ,1981), Potential function (Pek lenik ,1984) were also applied for forming part families .

Theoretically a good GT part family formation technique should have the following capabilities:

(1) Classify existing part into part families
(2) Classify new product into existing part families (if any).
(3) Effective application for different usages.

The function of most current GT part family formation techniques is to group existing components / parts into part families (King 1980, Mutel.et.al 1986).These methods cannot classify a new product into existing part families without reprocessing the entire data. The Multi objective cluster analysis method, the fuzzy cluster analysis method and some pattern recognition methods are the part family formation methods which have the ability of dynamic classifying.

5.3. Pattern Classification for GT: [50]

The area of (PR) usually deals with the automated extraction of information from signals, like waveforms, images and other multidimensional data. Since the GT classification codes can be treated as multidimensional data. It is suitable to use some of the PR techniques for GT application.

The word "Pattern" in Pattern recognition (PR) domain can be classified into two categories:

(1) Abstract Pattern : examples are ideas and arguments

(2) Concrete Patterns: this includes the patterns like symbols, characters, biomedical images etc.

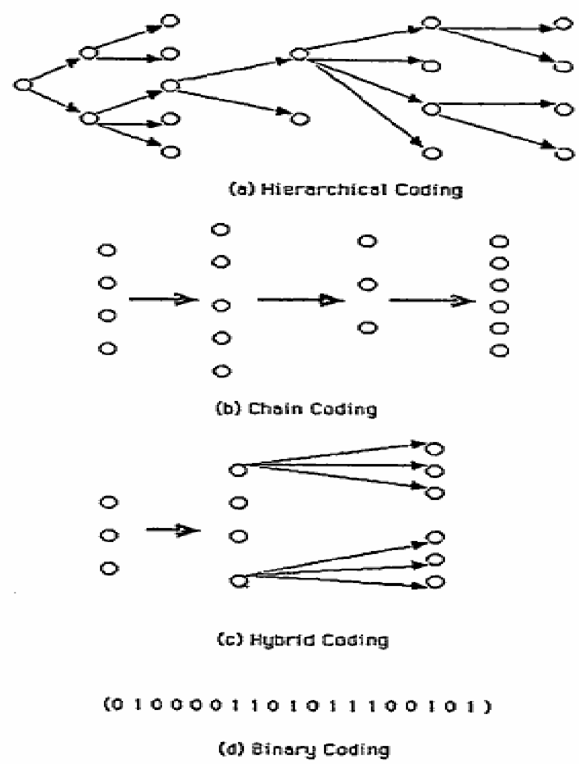


Fig: 5.1. Basic structure of GT classification and coding system [50]

A GT classification and coding system, or a classification scheme is a sample of concrete patterns. Pattern recognition can be loosely defined as follows:
"Pattern recognition is concerned primarily with the description and analysis of measurements, taken from physical or mental processes." (FU, 1980).
Conventional PR techniques are based upon two major methodologies: statistical and structural. They have been successfully applied on fields like weather forecasting, recognition of hand printed characters, speech recognition, medical analysis etc. Current developments in this field are using AI methods to solve variety problems. According to Bow (bow, 1985) the entire task of pattern recognition can be divided into three phases (fig: 2). they are data acquisition, data processing and decision classification.

(1) Data acquisition: Data from the physical world are gathered through a transducer and converted to digital format suitable for computer

processing. In this stage the physical variables are converted to a set of measured data.

(2) Data processing: Converting a set of discrete data into a mathematical pattern, so that those data are more suitable for computer analysis. The output of this phase is a pattern vector in pattern phase.

(3) Decision classification: a classifier which is in the form of a set of decision functions.

Using the Pattern recognition technique for GT classification and part family formation, the task of each phase can be described as:

(1) Construct a classification scheme which can be applied to extract the geometrical and technological features of the manufacturing parts using set of data/codes vectors.

(2) Elicit these features which are significant for GT part family formation.

(3) Apply a GT part family formation technique, a classifier to group the parts into part families.

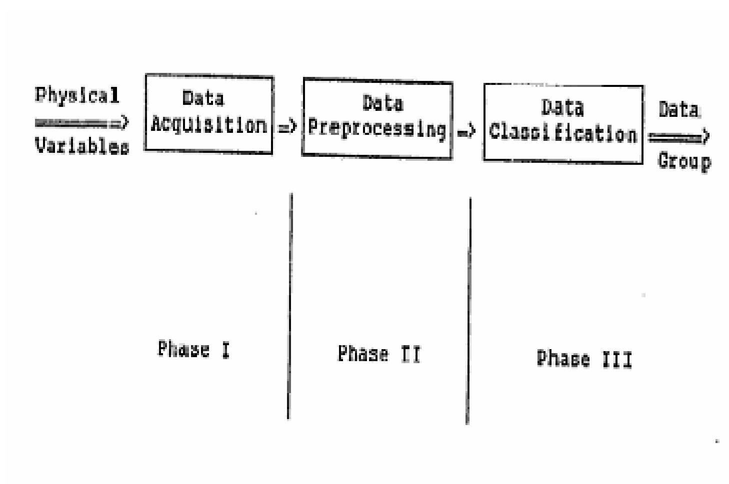


Fig: 5.2. Showing three phases of pattern Recognition [36]

5.4. Pattern recognition: [37]

Pattern recognition (also known as classification or pattern classification) is a field within the area of vision systems and can be defined as "the act of taking in raw data and taking an action based on the category of the data”. It refers to automatic transformation of data x_i (observations) into a set of symbols Ω_i (classes). Two-dimensional (2-D) object classification has several applications in addition to character recognition. These include airplane recognition, recognition of mechanical parts and tools, 13l and tissue classification in medical imaging (34). Several of the feature extraction techniques described in this chapter. These pattern classification techniques most successfully used in OCR till date.

In pattern recognition the features could be extracted from images in two ways:

(1) Using Gray Scale Images

(2) Using Binary Images

(1) Gray scale Images: [3] - A major challenge in gray-scale image-based methods is to locate candidate character locations. One can use a locally adaptive binarization method to obtain a good binary raster image and use connected components of the expected character size to locate the candidate characters.

However, a gray-scale-based method is typically used when recognition based on the binary raster representation fails, so the localization problem remains unsolved for difficult images. One may have to resort to the brute force approach of trying all possible locations in the image. However, one then has to assume a standard size for a character image, as the combination of all character sizes and locations is computationally prohibitive. This approach cannot be used if the character size is expected to vary.

(2) Using Binary Images: [3] - A binary raster image is obtained by a global or locally adaptive binarization of the gray-scale input image. In many cases, the segmentation of characters is carried out simply by isolating connected components. However, for difficult images, some characters may touch or overlap each other or other print objects.

Another problem occurs when characters are fragmented into two or more connected components. This problem may be alleviated somewhat by choosing a better locally adaptive binarization method, but (Trier and Taxt) have shown that even the best locally adaptive binarization method may still not result in perfectly isolated characters.

The binary raster representation of a character is a simplification of the gray-scale representation. The image function $Z(x, y)$ now takes on two values (say, 0 and 1) instead of the, say 256 gray-level values. This means that all the methods developed for the grayscale representation are applicable to the solid binary raster representation as well.

5.4.1. Template matching: [3] - In binary template matching, several similarity measures other than mean square distance and correlation have been suggested. To detect matches, let n_{ij} be the number of pixel positions where the template pixel x is i and the image pixel y is j, with $i , j \ \varepsilon(0,1)$;

$$n_{ij} = \sum_{m=1}^{n} \delta_m(i , j) \text{ -- } (1)$$

Where

$$\delta_m(i, j) = \begin{cases} 1 & if \ \ (x_m = i)\Lambda(y_m = j) \\ 0 & \ \ \ \ \ \ \ otherwise \end{cases} \text{ ----------------------------------- } (2)$$

$i , j \ \varepsilon(0,1)$; and y_m and x_m are the mth pixels of the binary images Y and X which are being compared.

Tubbs evaluated eight distances and found the Jaccard distance d_j and the Yule distance d_y to be the best:

$$d_j = n_{11} /(n_{11} + n_{10} + n_{01}) \text{ -- } (3)$$

$$d_y = (n_{11} * n_{00} - n_{10} * n_{01}) /(n_{11} * n_{00} + n_{10} * n_{01}) \text{ -------------------------- } (4)$$

However, the lack of robustness regarding shape variations mentioned in Section 2 for the gray-scale case still applies. Tubbs tried to overcome this problem by introducing weights for different pixel positions m and equation (1) will be replaced as: $n_{ij} = \sum_{m=1}^{n} p_m(k / i)\delta_m(i , j)$ --- (5)

This produces more flexibility and reduces error to a certain grade. So multiplying by weight m at each pixel position may revert the condition of failure up to some extent. So we are using this methodology even in our code to get the Binary template.

5.4.2. Steps in pattern recognition: [3]

(1) Gray-level scanning at an appropriate resolution, typically 300-1000 dots per inch.

(2) Preprocessing :

(a) Binarization (two-level thresholding), using a global or a locally adaptive method;

(b) Segmentation to isolate individual characters;

(c) (Optional) conversion to another character representation (e.g. skeleton or contour curve);

(3) feature extraction;

(4) Recognition using one or more classifiers;

(5) Post-processing.

As we shown the generalized steps in pattern recognition, though this could be varied according to the definition of the problem. For e.g. in our problem during raster scanning, we convert the image into binary template i.e. the area marked by mouse will be treated as 1 while the area remains blank will be treated as 0. Along with this co-ordinates of the points where mouse has traced will also be recorded. These are further used for pattern matching, as they act as the decision criterion in level: 2 in the hierarchy process.

5.4.3. Flow of Data in Pattern Recognition:

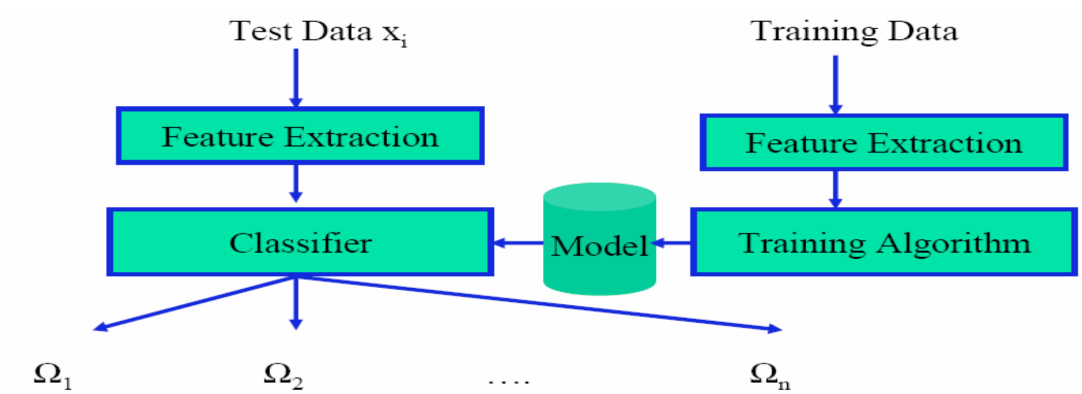


Fig: 5.3. Featuring Data flow for PR [5]

5.5. The methodology for automated retrieval of similar parts [16]

The objective of the problem at hand is to identify existing designs among the part databases, which are similar to a candidate part at the design stage. The similarity here is in one or more part characteristics, as specified by the designer. A two step methodology has been developed for the above. The first step is a search and retrieval procedure that acquires search intent information by capturing the 2D binary template from the Visual Display Unit (VDU) and identifies a set of potentially useful similar parts. The second step involves a more detailed similarity analysis of the similar parts identified by the first step. This step systematically sorts or ranks the similar parts according to their similarity to the candidate part; once again, the similarity request is specified by the designer. This is achieved using the opitz coding.

This two-step procedure is adopted for the sake of efficiency. The first step rapidly identifies a potentially useful part based on the criteria defined in the Level: 2 of Analytical Hierarchy process**.** The second step calculates detailed similarity based on critical design information only for the identified part from the first step. Thus the designer can review only the most relevant part and reduce time required for screening designs, process plans and production histories. The flowchart showing the overview of the methodology is shown in Fig. 1. The summary of the two steps of the overall methodology are also presented as follows:

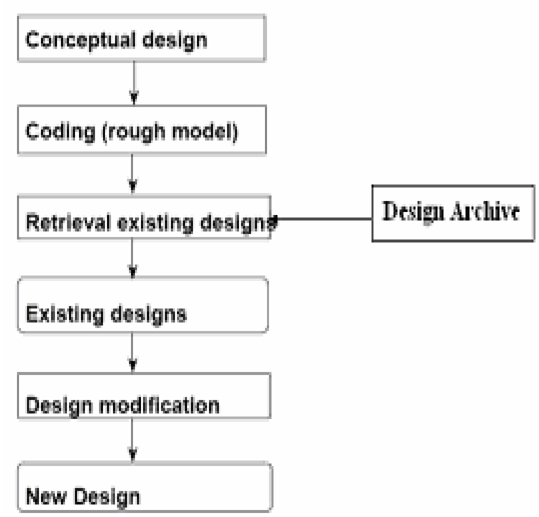


Fig: 5.4. Flowchart showing the overview of methodology [34]

5.5.1. Search procedure using GT coding scheme: [16]

This procedure uses the structure of the GT coding scheme to provide attributes (each corresponding to a specific part characteristic) with respect to which the user specifies the search intent and desired level of similarity. The procedure retrieves parts that satisfy this similarity level desired when compared to the candidate part.

This procedure could be applied on manufactured parts only, i.e. similarity ranking through visual inspection as we have shown earlier in chapter 5.

5.5.2. Procedure using Binary image for part classification: [16]

This procedure uses the binary vector generated after raster scanning, based on specific geometric and other design characteristics, to compute the numerical similarity between individual characteristics of the candidate part and a similar part. These numerical similarity values are integrated into a unique global measure through a formal combination method, and this global measure is used to sort the similar parts. The designer can view the similar part on the basis of the search results i.e. the File no on the screen with which the part is showing the similarity. The sorting step using bubble sort will just arrange the part with highest SI in descending order.

5.6. Product data information [16]

The overall methodology is applied for the selection of Bushings used as post insulators for transformers, the production of which involves complicated decisions related to both mechanical and electrical manufacturability concerns. A Transformer bushing is a multi-layered electrical product comprised of a layer of ceramic material, forming an insulation layer, and a complex mechanical ground plane. Bushings carry both surface mount and thru-hole components, which are mounted directly to the traces and pads of the Tank body, a thin conductive metallic layer. The substrate layer is a complex mechanical part that includes features such as holes, slots, chamfers and cutouts.

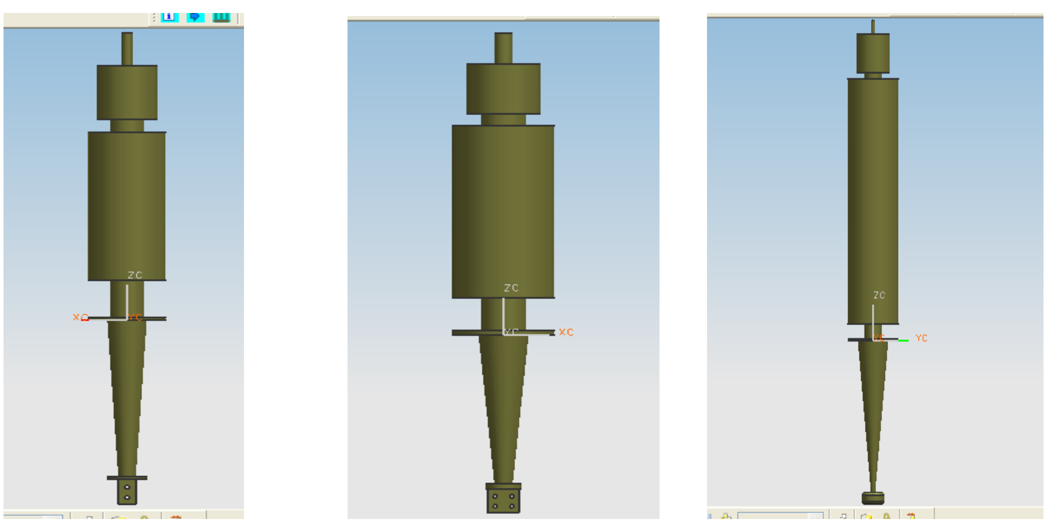


Fig: 5.5. Bushing bmp for product data information [Application UG]

Above is the modeled bush about whose features we are talking about. The operation of the search module is dependent on the availability of product data information of the bushings in two forms: (1) GT codes for the mechanical and electrical characteristics; and (ii) critical design information of the product.

5.7. Search procedure using the GT coding scheme [16]

The search procedure uses the concept of GT to establish similarity, either exact or partial, between the part under consideration (candidate part) and the other existing parts in the partner databases. The designer specifies the characteristics of the product that the search is to be based upon as well as the degree of similarity. These characteristics follow the same structure as the GT coding scheme. The search procedure processes the information given by the designer to identify the ranges (or sets) of digit values satisfying the similarity criteria, and uses these ranges/sets for the search. The end result of database retrieval from multiple patterns is a part similar to the candidate part with respect to the characteristics specified by the designer. A flowchart describing the search procedure employing GT codes is presented in Fig.6.

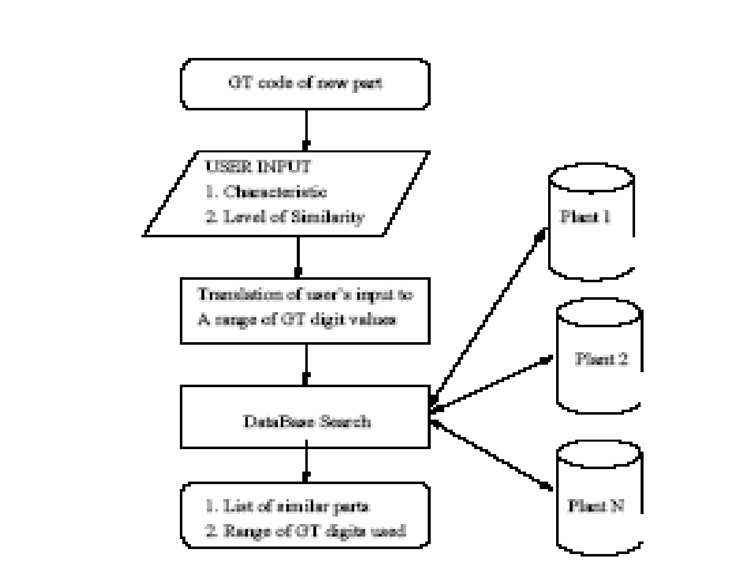


Fig: 5.6. Search procedure employing GT codes [16]

As a part of this work, the GT coding schemes were analyzed digit by digit for their representational significance and similarity relationships among the values of each digit were developed. The similarity level between a pair of values of a GI digit (or characteristic) (i, j) is assigned a numerical value called a **similarity index value** (SI[V.sub.ij]). According to the convention adopted the similarity index value has a numerical value between 0 and 1, where 0 stands for no similarity, and 1 stands for an exact match. Characteristics are classified into two types depending on the type of similarity index values the characteristic can be assigned: binary valued characteristics and continuous valued characteristics, which are described as follows.

5.7.1. Binary valued characteristics: [52]

Binary valued characteristics have been so called due to the binary nature of similarity measure that they assume. The similarity indexes can have only binary values {0, 1}, i.e., 0 for no similarity and 1 for an identical match.

The features (or values) of this class of characteristics (or GT code positions) are unique and no consistent similarity can be identified among them. Therefore, if i and j represent a pair of features of such a characteristic, SI [V.sub.ij] = 0 for i [not equal to] j, and SI [V.sub.ii] = 1.

Consider the example of the characteristic, mechanical function, which is represented by the GT code digits 5 and 6. The various digit values (00 to 99) represent various mechanical functions, like support, bracket, lever, screw, etc. It is difficult to assign a consistent partial similarity measure between the different digit values and the only similarity that can be consistently defined is one of exact match, all the others are considered to be totally dissimilar.

5.7.2. Continuous valued characteristics: [52]

The similarity indices for these characteristics can assume values in the closed interval [0, 1]. The features (or values) of this class of characteristics (or GT code positions) bear a decisive resemblance with each other, and the magnitude of the similarity index signifies the degree of resemblance or design/manufacturing similarity. Therefore, if i and j represent a pair of features of such a characteristic, the higher the similarity between i and j, the higher the value SI[V.sub.ij] is; note that SI[V.sub.ii] = 1. There are three types of continuous valued characteristics: primary feature characteristics, column feature characteristics and range feature characteristics. In the case of primary feature characteristics and column feature characteristics the rules for, similarity between features are used to calculate similarity index values between those features. These are used to develop a similarity index matrix which is used in the search process.

5.8. Concluding Remarks:

The above chapter provides us with an overview that in which manner features can be retrieved from the parts and how pattern recognition is connected with Group technology. We have also seen that how tubs had increased the accuracy of a binary vector conversion by multiplying a factor.

These methods require a decision making process which could be used to sort the binary images from the database to calculate the similarity index. As mentioned earlier that AHP could be one of the simplest methods to get the result from the selected criteria's. In the next chapter we would discuss and solve a full numerical problem to find out the similarity Index using AHP for part recognition.

CHAPTER 6

AHP APPLICATION IN PATTERN RECOGNITION & PART CLASSIFICATION

6.1. PROPOSED METHODOLOGY OF AHP: [32]

The analytic hierarchy process (AHP) possesses certain characteristics that make it a useful tool for making similarity decision in Group Technology as well as in Pattern recognition. The AHP's capabilities include: participatory decision making, problem structuring and alternative development, group facilitation, consensus building, fairness, qualitative and quantitative information, conflict resolution, decision support, and preferences structuring. The ability of the AHP to incorporate the human dimension (subjective preference) and to aid group decisions of choice, makes it a new favorable choice in the present scenario where we have much more sophisticated methods like Neural Network , Genetic Algorithm, Fuzzy logic & SMART etc. (D.L. Schmoldt et al, the Analytic Hierarchy Process in Natural Resource and Environmental Decision Making, pp- 289–305.)[11]

It requires paired comparison judgments concerning the dominance of one element over another for each of *n* elements with respect to an element on the next higher level using a 1-9 scale. This scale was given and proposed by (Satty.T in 1990) [35].

The paired comparison judgments are entered in a square matrix *A* of dimension *n*. For example, if an element i, is judged to be *moderately important* by comparison with element *j* with respect to the common element on the next higher level, a 3 is entered as value for the paired comparison judgment a_{ij} in the matrix *A* while the reciprocal value is entered for the paired comparison judgment a_{ji} .

Saaty defines a matrix *A* to be consistent, if the following condition is satisfied

$$a_{ik} * a_{kj} = a_{ij} \qquad \forall \quad i, j, k = 1.................n \qquad .... (1)$$

For example, if $a_{23} = 2$ and $a_{34} = 3$ are fulfilled, in a consistent matrix the value of a_{24} has to be 6. The essential idea of the AHP is that a matrix *A* of rank *n* is only consistent if it has one positive eigenvalue $\lambda_{\max} = n$ while all other eigenvalues are zero. Further, Saaty developed the *consistency index* (*CI*) to measure the deviation from a consistent matrix:

$$\text{CI}= (\lambda_{\max} - n)/(n-1) \text{ .... (2)}$$

The *consistency ratio* (*CR*) is introduced to aid the decision on revising the matrix or not. It is defined as the ratio of the *CI* to the so-called *random index* (*RI*) which is a *CI* of randomly generated matrices

$$CR = CI / RI \text{ .... (3)}$$

So the above prevails us with a brief review of what I am going to applied for Pattern recognition in Group Technology.

Now as per the application of AHP for pattern recognition a sequential flowchart is shown, which provides a step by step procedural sequence, that how AHP could be applied for the existing problem.

Mainly what we have, done is that first converting the image to binary pattern. Now those patterns are stored. As per the relative table given by Satty, a random matrix will generate for the predefined characteristics .Using which further weights will be generated. This could then be differentiated on the ranking parameters given by saaty to match the similarity of the existing image with previous patterns.

6.2. Steps in Applying MCDM model (AHP):

(1) Establish decision box refers to the brain storming session .In this section we have decided the decision parameters on which comparison to be made.

(2) Identify the alternatives refers to the Level: 3 i.e. the source of compare. In a bit simplified manner, we can say that alternatives are basically the objects with which we have to compare the input data. If we talk here in terms of classifier, then it could be interpreted as: that these the trained for the existing patterns and using this only, we have to check the similarity Index of the incoming part. This SI would be the only source in deciding the Family for the part in Group technology.

(3) Identify the criteria deals with level: 2. In this step one has to decide the features or classifiers or criteria's on the basis of which , we will made pair wise comparison as stated by Satty. Basically these criteria's are the only keys on the basis of which random matrices are generated.

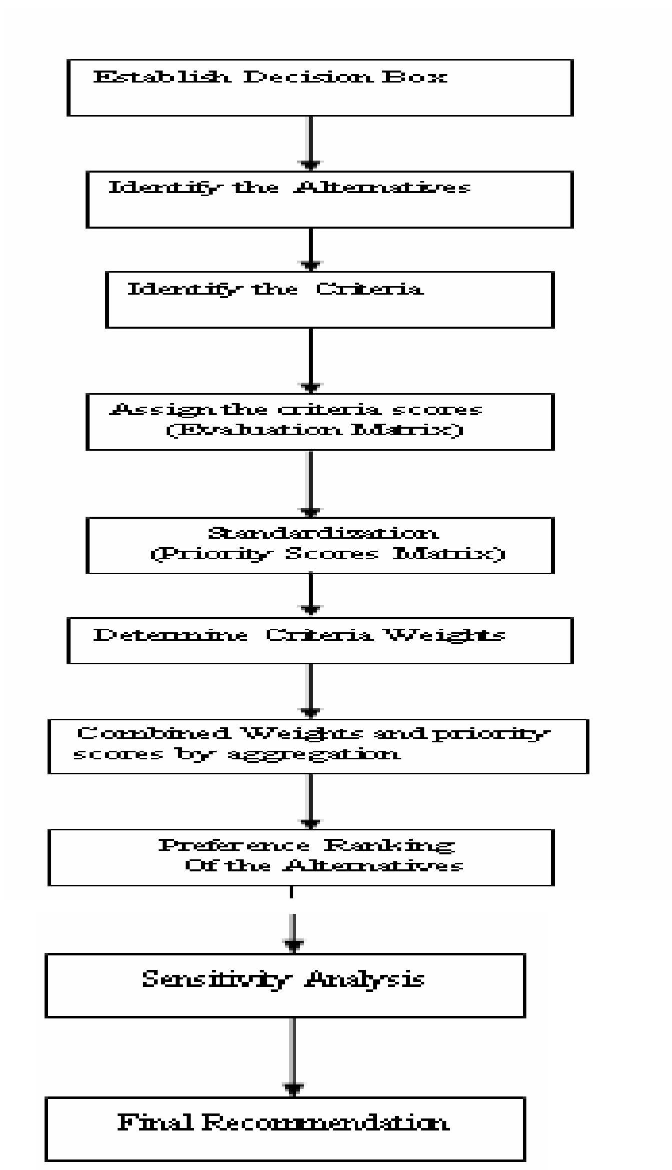


Fig: 6.1. Steps in applying a MCDM model (AHP) [53]

(4) As I have discussed above that for different criteria's, Evaluation matrices are generated randomly as per the decision criteria defined by satty, according the priority of one with respect to the other.

(5) Standardization matrix, i.e. once the matrix has been generated than we have to find the priority vector for each criterion. The evaluation matrix with these priority vectors is called standardization matrix.

(6) Now form priority vector using (Subjective, Objective, Geometric mean or Integrated method) of the methods find the weight for the Global stiffness matrix. Subsequently these weights are also determined for Local stiffness matrix, taking a pair of two each time.

(7) Now both local and global matrices are combined to obtain a aggregation matrix. This matrix will be further used for ranking purpose. Though its main evaluated function will be as follows:

Let v(y) be a function of both i.e. weight vector for local as well as for global, than it could be defined for prefence ranking acc. to the formula shown below

$$v(y) = \sum_{i=1}^{q} w_i y_i \text{ -- (5)}$$

(8) From the above results find out (Consistency Index) CI for the above matrix ,as shown above , for ranking and sensitivity analysis.

(9) Sort the following acc to the highest weight coefficient for the greatest similarity index and print out the result in the form of % similarity with the existing parts, and optiz coding will decide the part family classification for group technology.

6.3. Application of AHP in the Classification:

Analytical Hierarchy Process is used here for ranking and finding the similarity index of the newly designed product with the previously stored in the library. That is, it will serve the two purposes (1) Pattern recognition for part classification (2) Ranking of parts similarity on the basis of Similarity Index (SI) or Consistency Index (CI). This will be used in the UG /NX 3.0 environment. Its main purpose is that user can identify the part, i.e. could able to find that the whether the newly ordered part matches with previously made any of the parts, if yes? , than up to what extent. Mean to say that if the similarity is above the minimum modifiable limit than user need not to make the new part but he can customize the existing part to the new one. This will not only increases the efficiency but also reduces the lead time drastically and increases the overall production.

6.4. Steps in our Study:

For AHP the most important thing is comparison matrix. For that from the figure, we can say that the things we required on priority basis are the characteristics or features. These are finally taken as base objects for pair wise comparison in AHP.

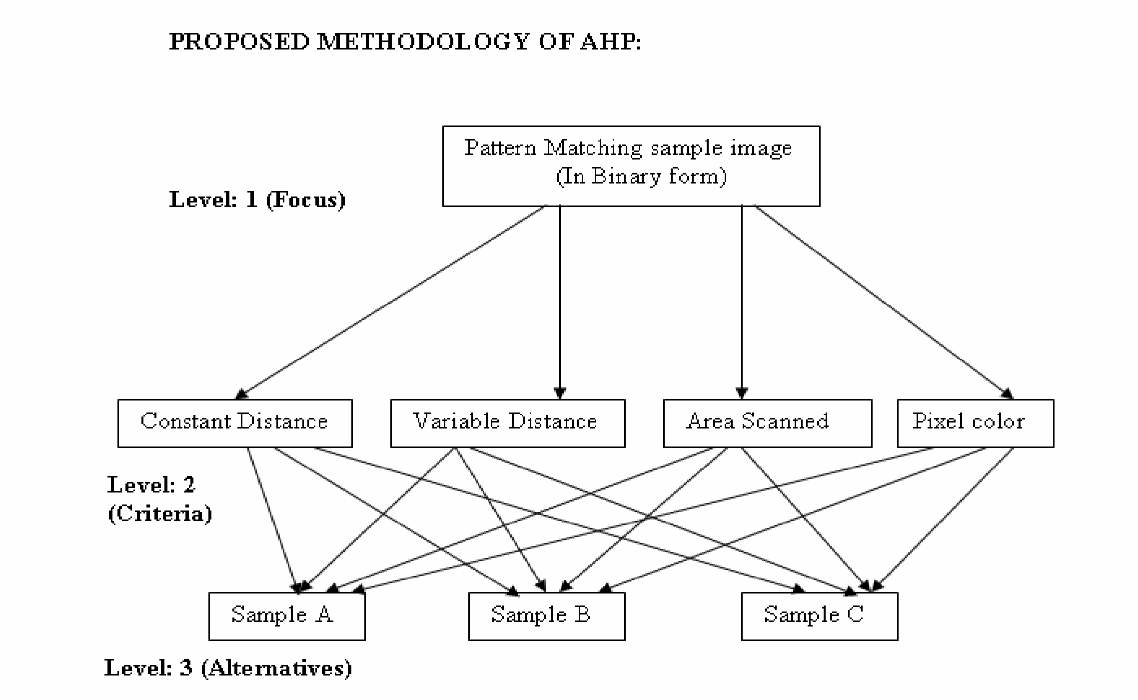


Fig: 6.2. Proposed hierarchy model for Pattern recognition problem

6.4.1. Selection of Criteria:

In the problem Constant distance, Variable distance, Area and Pixel color as taken as four parameters. The thing is why these parameters are chosen? For this during pattern recognition literature survey (Oivind Due Trier, 1995) suggested that "The requirements of a good feature extraction method make selection of the best method for the given application a challenging task". He stated that a locally adaptive binarization method to obtain a good binary raster image and use connected components of the expected character size to locate the candidate characters while a gray scale based method is typically used when recognition based on the binary raster representation fails, so the localization problem remains unsolved for typical images.

A binary raster image is obtained by a global or locally adaptive binarization (D.Tier & T.Taxt) of the gray scale input image. In the present problem the first phase is to convert the gray image into a binary pattern. The binary raster representation of a character is a simplification of gray-scale representation. During scanning for binary image some parameters are required which could further be used for part recognition. These parameters help in overcoming that problem of failure in Binary raster representation.

Level 1: refers to the new design displayed on the screen .That is we can explain it as a new model, ordered by the customer .So it is the input from which we will get the image as well as values for all other criteria's, like Variable distance co-ordinates, Constant distance co-ordinates, Area of the scanned image and pixel pattern for matching the one in database.

Level 2: Here we will store all the scanned values as well as the binary pattern of the image. This data will further be used for template matching after assigning the priority ranking according to the Saaty relative matrix. Further we will discuss it using a Numerical example, though in code these will be selected randomly using a function Srand.

Level 3: It stores the binary template of all the samples one have teaches to the software.It could be compared as a database of a Expert sytem or a library of templates for pattern recognition, where each sample has been stored whose weights has been modified. This is destination point i.e. what all we have taught to the computer or to the software can be analyzed at any moment from this point.

6.4.2. Application of AHP to get Ranking for parts and weights for classifier:

The procedure I have followed to solve the problem is as follows:

(1) Define a Pair-wise comparison matrix.
(2) Find out the priority vectors and assign them to each criterion.
(3) Set attributes to pattern array.

(4) Again find priority vectors for local stiffness matrix.

(5) Find local weights for the No. of patterns in the database.

(6) Combine local and global matrices to get the modified weights.

(7) Find product of weight in step 5 with the local priority vectors in step 1

(8) Rate the values of A, B, C…….. I.e. no of sample patterns.

(9) Find $\lambda_{\max}$,CI, ACI , Normalized vectors and w for entire patterns.

(10) Compare them to find the similarity ratio with the input image.

(11) Display the result, i.e. the % matching with the model in database.

In Table: 1. four criteria for selecting a pattern are compared in a pair wise fashion and assigned a relative importance score.

**

	Constant Distance	**Variable Distance**	**Area**	**Pixel color**
Constant Distance	1	1	1/3	5
Variable Distance	1	1	1/2	9
Area	3	2	1	3
Pixel color	1/5	1/9	1/3	1

Table: 6.1 - Scale of relative parameters for PR

Now the second step is to find the relative weights of the criteria (priorities). Now in the simplest way these can be computed as the normalized geometric means of the rows (which are very close to the eigen vector corresponding to the largest eigenvalue of the matrix).

The priorities by the geometric mean can be traced out using the relation shown below. The geometric mean minimizes the logarithmic error:

$$\sum_{i=1}^{n}\sum_{j=1}^{n}(\ln(a_{ij})-\ln(p_i/p_j))^2 \text{ -------------------------------- } \textbf{(a)}$$

The Geometric means are computed as:

t1 = $\sqrt[4]{1*1*1/3*5}$ -- (1)

t2 = $\sqrt[4]{1*1*1/2*9}$ -- (2)

t3 = $\sqrt[4]{3*2*1*3}$ --- (3)

t4 = $\sqrt[4]{1/5*1/9*1/3*1}$ -- (4)

So, the relative weight (priorities) of the criterion research is obtained as

p1 = $t1 \;/\; (t1 + t2 + t3 + t4)$ ------------------------------------- (5)

** Global matrix for priority classification.

p2= $t2 \;/\; (t1 + t2 + t3 + t4)$ ------------------------------------- (6)

p3= $t3 \;/\; (t1 + t2 + t3 + t4)$ ------------------------------------- (7)

p4= $t4 \;/\; (t1 + t2 + t3 + t4)$ ------------------------------------- (8)

Now to find the $\lambda_{\max}$ we can solve it as follows:

$\det[A - \lambda I] = 0$ -- (9)

That is from above if:-

$$A = \begin{bmatrix} 1 & 1 & 1/3 & 5 \\ 1 & 1 & 1/2 & 9 \\ 3 & 2 & 1 & 3 \\ 1/5 & 1/9 & 1/3 & 1 \end{bmatrix}$$ -- (10)

Then

$$\text{Det}\,[A - \lambda I] = \det\begin{bmatrix} 1-\lambda & 1 & 1/3 & 5 \\ 1 & 1-\lambda & 1/2 & 9 \\ 3 & 2 & 1-\lambda & 3 \\ 1/5 & 1/9 & 1/3 & 1-\lambda \end{bmatrix}$$ --------------------- (11)

Solving the equation (11) we can get the value of $\lambda_{\max}$. To obtain the eigen vector corresponding to the first eigenvalue

Aw = $\lambda_{\max}$ *w;---(12)

Solving it in a same manner we can get:

$$\begin{bmatrix} 1 & 1 & 1/3 & 5 \\ 1 & 1 & 1/2 & 9 \\ 3 & 2 & 1 & 3 \\ 1/5 & 1/9 & 1/3 & 1 \end{bmatrix}\begin{bmatrix} w1 \\ w2 \\ w3 \\ w4 \end{bmatrix} = \lambda_{\max}\begin{bmatrix} w1 \\ w2 \\ w3 \\ w4 \end{bmatrix}$$ ------------------------ (13)

From matrix theory it is known that a small perturbation of the coefficients implies a small perturbation of the eigen values. Hence we still expect to find an eigen value close to n, and select the elements of the corresponding eigenvector as weights. It can be proved that

$\lambda_{max} \geq$ n;

And the matrix A is consistent if and only if λ_{max} = n. A way of measuring the consistency of the matrix A is by defining the *consistency index* (CI) as the negative average of the remaining eigenvalues:

$$CI = \sum_{\lambda<\lambda_{max}/n-1} \lambda = \lambda_{max-n}/n-1 \text{-------------------------------- (b)}$$

Now from equation (13)

w1 + w2 + (1/3)*w3 +5*w4 = λ_{max} *(w1) ------------------------------- (14)

w1 + w2 + (1/2)*w3 +9*w4 = λ_{max} *(w2) ------------------------------- (15)

w1 + w2 + *w3 +3*w4 = λ_{max} *(w3) ------------------------------------- (16)

(1/5) w1 + (1/9)*w2 + (1/3)*w3 +w4 = λ_{max} *(w4) -------------------- (17)

Finding the values of w1, w2, w3, & w4. Finally normalize W by equating its coefficients total to unity. So, now the priority matrix will be:

	Constant Distance	**Variable Distance**	**Area**	**Pixel color**	**Priority Value**
Constant Distance	1	1	1/3	5	.2632
Variable Distance	1	1	1/2	9	.337
Area	3	2	1	3	.4774
Pixel color	1/5	1/9	1/3	1	.217

Table: 6.2 - Priority Vectors for Global matrix

So, after finding the global priority matrix now we have to compare the alternatives on each criterion, and trace out the priority values for local stiffness matrix.

Constant Distance	A	B	C	Priority
A	1	4	4	.65108
B	x	1	3	.2355
C	x	x	1	.11338

Table: 6.3 Pair wise comparison for constant distance

Variable Distance	A	B	C	Priority
A	1	1/9	1/3	.0856
B	x	1	1/4	.3329
C	x	x	1	.5794

Table: 6.4 Pair-wise comparisons for variable distance

Area Scanned	A	B	C	Priority
A	1	1/2	1/4	.1495
B	x	1	1/2	.3763
C	x	x	1	.11338

Table:6.5 Pair-wise comparisons for Area

 # x refers to fulfill the criteria that $a_{ij} = 1/a_{ji}$ *and* a_{ij} *>0.*

 # Matrix should be consistent and invertible.

Pixel color	A	B	C	Priority
A	1	1/2	1/2	.1958
B	x	1	1/2	.3108
C	x	x	1	.49325

Table: 6.6 Pair-wise comparisons for pixel color

Here finally there are all the four local as well as global priority matrix. Now combine the local and global priority matrices to get the final rank order. This will provide us with the part with maximum similarity index within all the existing samples.

6.4.3. Local and global priorities vector:[20]

Satty with R.Nagi in 1980 proposed that combined weight matrix can be solved by using:

$$A = \sum_{i=1}^{n} \sum_{j=1}^{n} t_i * w_{ij(local)} \quad where \quad \forall\ i,j\ \ w_{ij} > 0 \text{---------------------(c).}$$

After Nagi, other methods were also proposed by some geniuses like Chu.et.al suggested the least square method for finding the weight, while two other more accurate methods which are used prominently now days are Subjective and objective approaches [Decision making processes, 1980, Nagi.R]. But both of them has some disadvantages as we have shown earlier in chapter 7, so here we are going to drive weights using new approach of integrated weights.

6.4.4. Integrated Approach: [20]

Surveying both objective and subjective approach, we find that in the attribute Pair-wise comparison matrix D=$[d_{kj}]_{n*n}$ the weights determined by the above subjective approach reflect the subjective consideration of the DM. In the objective decision matrix A= $[a_{ij}]_{m*n}$ the weights determined by the above objective approach reflect the objective information.

In order to make weights reflect both subjective and objective factors, we integrate the subjective approach (i.e. model (e)) and the objective approach (i.e. model (k)) to set up the following two-objective programming model:

Minimize: $\left[\begin{array}{l} Z_1 = w^T F w \\ Z_2 = w^T H w \end{array} \right]$ ---(r)

Subject to $e^T w = 1$ and $w >= 0$

In this case Lagrange's function would be

$L_3 = w^T Q w + 2\lambda_3 (e^T w - 1)$ ....(s)

and from differential equation solution

$w^* = Q^{-1} e / e^T Q^{-1} e$ .... (t)

$\lambda^*{}_3 = -1 / e^T Q^{-1} e$ .... (u)

Where w^* is the weight vector determined by the subjective and objective integrated approach. Its accuracy is greater than above two as it is a hybrid of the two.

** For proof and derivation of Integrated refer previous chapter.

6.4.5. Result: Final weighting coefficient of Rank Matrix

So presently the weights have been calculated using integrated approach which provides us more accurate and justified weights for pattern recognition.

	Constant Distance(1)	Variable Distance (2)	Area (3)	Pixel color (4)	Weights(w)
	.263	.337	.477	.217	
A	.65108	.0856	.1495	.1958	**.312**
B	.2355	.3329	.3763	.3108	**.419**
C	.11338	.5794	.11338	.49325	**.556**

Table: 6.7 – Combined rank table considering all patterns

$$(.263)*\begin{bmatrix}.65108\\.2355\\.11338\end{bmatrix}+.(337)*\begin{bmatrix}.0856\\.3329\\.5794\end{bmatrix}+..........(.217)*\begin{bmatrix}.1958\\.3108\\.4932\end{bmatrix}=\begin{pmatrix}.312\\.419\\.556\end{pmatrix}\begin{bmatrix}A\\B\\C\end{bmatrix}$$ **--------- (18)**

So, the above result shows the "C" part shows the maximum similarity for the current model, i.e. model C can be customized for the new existing part.

6.5. Concluding remarks:

The above chapter shows the detailed analysis as well as fully solved numerical method for pattern recognition approach. The result will show the % and the part similar to the existing one on the screen. This all will be decided on the basis of the weights calculated for the final rank matrix as shown in equation 18.

This will prove as a very useful tool during designing a new part. An enhanced database from different companies shows that out of 100 times system founds a part with a similarity ratio of 80 and above. In the next chapter we are going to design the bush using KBE and Native UG. This is mainly used to create a part file library, which is required as an input to the system at later stage.

CHAPTER 7

CLASSIFICATION OF TRANSFORMER BUSHING

7.1. High Voltage Condenser Bushing [33]

Design and Automation of bushing is basically a module from the project "Complete design & Automation of a Transformer using Knowledge Fusion & Expert system".

This project mainly deals in developing Knowledge based software that could develop the Transformer design as well as a 3-D model for manufacturing purpose just by receiving some of the parameters from the customer, furthermore it not only produce programs for CNC m/c, but also helps user by providing the suitable data for deign a specific transformer for a special purpose, for e.g. the transformers used for household power transmission are totally different from the transformers being used in heavy Kaplan and Francis turbines for power generation.

1. Bushing: - Bushing is one of the most important component that are fitted to the electrical equipments like the transformer, the switchgear etc. It is an insulating structure for carrying the HV conductor through an earthed barrier.

The Bushings has to: -

1) Carry the full load current
2) Provide electrical insulation to the conductor for working voltage and for various over voltages that occur during service.
3) Provide support against various mechanical forces.

2. Classification of Bushings:
Bushings are classified according to the following factors

7.2. Applications or Utility [33]

(a) Alternator Bushing:

AC generator requires bushing up to 33 KV, but 22 KV is more usual. With modern alternators, current rating up to 20,000 A are required.

(b) Bushing for Switch Gear

In the switchgears Bushings are to carry the conductors through the Tank wall and support the switch contacts.

(c) Transformer Bushings

Transformer requires terminal bushings for both primary and secondary windings. In some cases a high voltage cable is connected directly to the transformer via an oil filled cable box .A Bushing than provides the connection between the cable box and Transformer winding.

(d) Wall or Roof Bushing

In the recent years many substations of 132Kv and above in unfavorable situations have been put inside a buildings are used.

(e) Loco Bushings

These bushings are used in freight loco and AC EMU Transformers for the traction application.

7.3. Non Condenser and Condenser Bushing:[33]

(a) Non-Condenser Bushing

In its simplest form, a bushing would be a cylinder of insulating material, porcelain, glass resin, etc. with the radial clearance and axial clearance to suit the electric strengths. The voltage is not distributed evenly through the material, or along its length .As the rated voltage increases the dimensions required become so large that this form of bushing is not a practical proposition. The concentration of stress in the insulation and on its surface may give rise to partial discharge. This type of Bushing is commonly used as low-voltage bushings for large generator transformers.

(b) Condenser Bushing

In this type the conducting cylinders are inserted in the insulation to divide the wall thickness into a number of capacitors .In this way the voltage distribution in the material and along its surface can be controlled.

7.4. Design of Bushing [33]

In its simplest form a bushing consist of a central conductor embedded in a cylindrical insulation material having a radial thickness enough to withstand the voltage. The Design of bushing depends on various voltages and over voltages that it has to come across during service. In order to study the factors, which influence the design of a bushing, it is convenient to consider a transformer bushing with one end in the air, and the other in oil. The important factors, which affect the design, are:

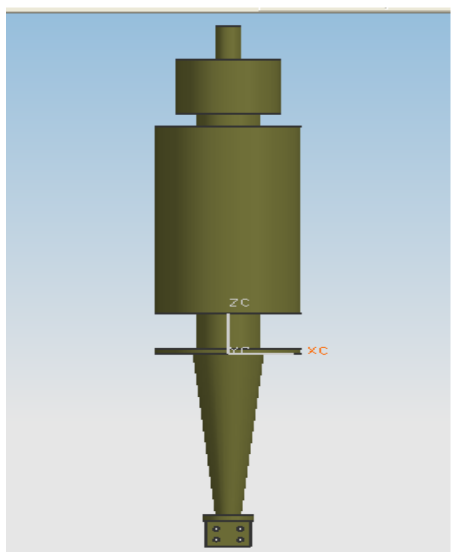


Fig: 7.1. Showing condenser bushing model [UG]

(a) Air end clearance

The air-end clearance has to be sufficient to meet the specified over-voltage tests. It is also determined by the creepage distance, and the proportion of it that is protected from the rain, having determined the air-end length, the air end dimensions of the internal condenser can be determined. It is not necessary to grade 100%. Internal grading of 70% or less will give adequate surface grading for large bushings.

(b) Oil end Clearance

An internal breakdown unlike air flashover is more severe, specifications therefore demand an internal breakdown with a sufficient margin (about 15%) above the air withstand value. Both power frequency, and impulse voltage withstand tests have been used to specify this characteristic.

(c) Number of Condenser Layers

The no of partial condensers are so chosen that the test voltage of each partial condenser should be between 10Kv to 15Kv .If more foils are introduced it would cause too many folds, complicating the manufacture of bushing of high voltage class.

(d) Length of Earth Layer

The length of earth layer of a bushing is usually determined by the accommodation required for current transformer, or by mounting considerations, though in some cases it may be allowed to assume its optimum dimensions in relation to the radial dimensions .The ratio of length of 1^{st} foil (L1) and length of n^{th} foil (Ln) may be taken between 3 to 4 .The ratio is denoted by α.

(e) Radial Gradient and Diameters

The radial gradient is limited by the necessity for avoiding damage by discharges at the power-frequency test voltages whether one minute or instantaneous. If the ratio of the earth layer diameter to that of the conductor (r^n / r^o) is denoted by β, the stresses at the HV end and the earth voltage end will be equal, if the product of α and β is unity.However it is not always possible to achieve this value. Hence α and β can be varied from 0.8 to 1.2. i.e.

$$\text{If } \alpha*\beta=1 \text{ then } L_n D_n = L1* D0.$$

(f) Equipotential Layer Position

After determining the dimensions of inner and outer layer of the condenser, the position of the other layer can be calculated .The basis of the design of the condenser bushing is generally equal partial capacitances, which means equal voltage on them and equal axial spacing between the ends of layers.

7.5. Construction Details and Main Parts of Bushing [33]

7.5.1. Core

The core of bushing consists of a hollow or solid metallic tube, over which high grade electrical craft paper is wounded. For condenser cores, conducting layers of metallic foil are introduced at predetermined diameters to make uniform distribution of electrical stress. The winding of the condenser core is done in a dust free chamber .The core is then processed; this comprises of drying in a high degree of vacuum (0.005mm) ,and then impregnating with high quality , filtered and de-gassed transformer oil.

7.5.2. Porcelain

Bushing for outdoor application are fitted with hollow porcelain insulators .The OIP bushings are provided with insulators ,both at air And oil ends ,thus forming an insulator envelope ,and the intervening space may be filled with an insulating liquid or another insulating medium.

The function of an insulator is to resist flash over in adverse conditions .this is determined by

(a) The profile of the dielectric
(b) The mounting arrangement of the insulator, i.e. vertical, Horizontal or inclined.
(c) The properties of the surface, i.e. hydrobhobicity, toughness etc.

For insulators used in air-end side, a minimum specific creepage distance is required for different pollution levels.

Based on the above information, the shape of porcelain sheds are:

(a) Plain
(b) Anti –fog
(c) Alternate

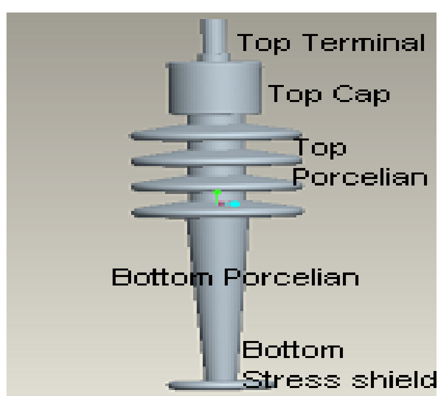


Fig: 7.2. Detailed description of the parts in a condenser Bushing [UG-APP]

7.5.3. Top Cap

This is a metallic housing for the spring pack .It serves as an inbuilt oil conservator to cater for oil-expansion, and has an oil level indicator. In many cases it also serves the purpose of corona shield.

7.5.4. Mounting Flange

This is used for mounting the bushing on an earth barrier, such as a transformer tank or a wall. It may have the provisions for the following:

(a) CT accommodation Length
(b) Rating plate giving rating and identification details of the plate
(c) Test tap
(d) Oil drain plug for sampling of oil
(e) Air release plug

The design of the flange and the top cap is such as to minimize the loss due to hystersis and eddy current effects. When heavy currents are being carried this loss raises the temperature of the flange and the top cap to a noticeable extent. For heavy currents ordinary cast iron materials cannot be used, hence non-magnetic materials such as stainless steel or aluminum are used.

7.5.5. Test Tap

The test tap is provided for measurement of the power factor and capacitance of the bushing during testing and service. The test tap is connected via a tapping lead to the last condenser foil of the core within the bushing. During normal service, this tapping is electrically connected to the mounting flange through a self-grounding arrangement.

7.6. Concluding Remarks:

Chapter gives us an overview of the designing and manufacturing of bushing. It shows total number of components and detail out functioning of each component with construction details. Now we are fully aware with all the design details and the stresses or the forces developed in insulators. So now for designing of bushing in UG we are going to use knowledge base engineering (KBE). In the next chapter modeling and assembly of bushing will be discussed with respect to transformers.

Chapter 8
Design & Integration of Bushing
(A Case Study)

8.1 Problem Formulation

In the above section as we have inferred about electrical stresses and bushing design, this shows that bushing plays a key role in minimizing the energy losses in the transformer. It still requires a much more concern to deal with electrical losses and minimize the losses in transformer.

In the present project i.e. "Design and Automation of Transformer "in all there are two types of bushings:

1. Condenser Bushings
2. Porcelain Bushing

1. This includes 40 and 38 types of bushings respectively based on their Voltage Ratings and type of Insulation. So the first and foremost problem is that to model 78 models of bushings would consume a lot of time ,so this has been resolved out using Part classification or feature recognition. The user requires that only 4-5 models for both types of bushings should be enough to generate all the models of bushings.

2. The second phase includes integration, i.e. coding has to be done in Uni-Graphics using "Knowledge Fusion" and "U-func" for designing the User Interface and for proper placement of bushing on the tank body as the case has been selected by the user in User Interface. This will include "Assembly Cloning" and developing "Quantitatively Synthesized Relations" for the proper placement of bushing.

8.2 Problem Identification and Module Formation

For modeling of bushing the first and foremost things we require is all the drawings and dimensions that are to be automatized. Bushings as we have discussed are mainly of two types Porcelain and Condenser respectively. Overall their are 78 drawings of bushings that have to be modeled.

One way is that first we model all the 78 bushes and than start their automation but from engineering point of view that would not be a solution, in fact:-

(1) That would make the software Bulky
(2) Would increase the processing the time of application.
(3) Solution does not fit within the time limits as prescribed by the user.
(4) Integration and automation of 105 models of bushing in a project where already 750 components have to be integrated would be not less than a heavy blow to the job.

Taking above factors into consideration an alternate solution is to be traced out which could fulfill all the requirements.

8.2.1. Literature Review

(a) Theswood in (1987) [29] from university of California also deals with a similar problem during classification of bearings in heading a project from General Motors .Their he suggested that the problem can be handled in a manner as we do in CIMS i.e. using "OPITZ" coding.

He suggests that a similar code could be developed by segregating the features from the drawing and than group them to make a specific class. Here from features he meant to say, to make distinction in model on the basis of components we have to design for e.g. if we have to model a plate than there are two variables Length and width. Controlling these two variables we can get different models of a plate which can be used at different places.

This was mainly a geometrical approach in which first all the parameters in the drawing are studied in depth, based on them a table of variables is to be prepared. Finally developing some relations he was able to get all the models of bearings, simply by making 12 classes based on the component features in bearings. He devised this type as "Component based Classification" (CBC).

(b) One more approach was also developed by "Gianni Caligiana" [14] on 3rd June 2005.During his work for designing testing rigs for composite materials to test different valves of scramjet engines.

Instead of developing separate rigs for each valves based on CBC he jot down the common in deigning a valve for e.g. Pressure plate, runner way, gaskets etc. He finds it easy to modify the rig just by varying the dimensions of these specific components.
This not only enhance the working speed but "Alnoca" company was able to complete the project of 5 years within a time span of 3 years which increased their profit by 15%.

8.2.2. Problem Review & Strategy Development

Intervening with the problem of bushing and based on the success rate of CBS, It is proposed to use the same technique for solving the problem. Different components of bushing have been detailed out in above section:

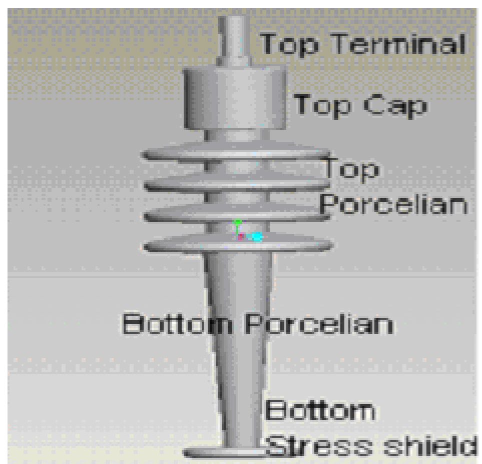


Fig: 8.1. Details of bush for finding the similarity [UG]

i.e. Top Terminal, Top Cap, Top porcelain, Drum, Flange, Bottom porcelain, Base plate ect. After studying the drawings in detail, finding says that in most of the cases either Base plate, Top Cap Diameter, Bottom Cap Diameter, Flange Thickness, Type of Porcelain and PCD of the flange varies. Finally three criteria's on the basis of which bushings can be classified or grouped are:

1. Drawing Number (Based on Base plate of the Bush).
2. Voltage Rating
3. Current Rating

1. Type of Base Plate: This component is there in bushing to connect the base wiring to the transformer. Basically only five types of base plates are there in all the models. These are Rectangular, Square, Circular, Conical and triangular. These types are based mainly on the stresses developed with increase in voltage and current

ratings. So one of the types is to be called based on calculations to reduce the stress.

2. Voltage Rating: Mainly I divided it in 5 categories i.e. HV, LV, LV1, LV2 and TV where LV refers to Low Voltage category, HV refers to high voltage category and TV refers to Tran's voltage category. In all they varies from 17.5 KV to 10000KV.

3. Current Rating: Similar is the case with current rating that with increasing the current rating Heat losses, stresses and skinning effect goes on increasing, so the dimensions should be varied in a similar proportion to these losses.

Though the criteria have been decided to select the bushings and later we will discuss that how all the 78 models can be made from 5 models using GT with Pattern Recognition.

Now the problem is with the platform which could provide us the flexibility to customize and integrate the models as per the problem requirement. After comparing the different existing software's, there is no doubt to say that Uni-Graphics provide the best platform to integrate and customize any of the designed models. Using Knowledge Based Engineering (KBE), Knowledge Fusion (KF), and U-Func –a special coding language where one can make his own DLLs to integrate the library, so that these could be used for further coding.

Using KF we decided to define the classes which further be used as relative conditions to distinguish the models in Bushing. The class coding to be defined in KBE is as follows:

```
#UG/NX:2 PART 1271 560 0 1 1 15 2400 2003100 00000741
For example:
# List of option menu for selecting Bushing Nomenclature
(List) ui_opt_bushing_nomencleature_list:
@{
        $list1 << {"HV"};
        $list1 << {"LV" , "IV" , "MV"};
```

```
        $list1 << {"LV1" , "IV1" , "MV1"};
        $list1 << {"LV2" , "IV2" , "MV2"};
        $list1 << {"TV" , "LV" , "SV"};

$return_list << if (ui_opt_select_bushing: = "HV") then $list1
                else if (ui_opt_select_bushing: = "LV") then $list2
                else if (ui_opt_select_bushing: = "LV1") then $list3
                else if (ui_opt_select_bushing: = "LV2") then $list4
                else $list5;
};
```

8.3. Integration of the Model with Transformer using Knowledge fusion:

As for automating the model and integrate the model with the assembly the Knowledge Fusion is required. "Knowledge fusion is a technology that allows an engineer to create a product model based on rules that capture the methodology used to design, configure and assemble products. Knowledge Fusion facilitates the capture of the intent behind the product design by representing the why and how in addition to what of a design."

So to update the expression and assemble the model in the transformer assembly and to create the Bill of Material of the model with entering the Unigraphics Native (Unigraphics Modeling Feature) the coding is required to be done in Knowledge Fusion. The coding is to be done using U-Func and Knowledge Fusion for achieving the same.

8.4. Modeling and Designing of Bushing

In this phase the model is to be developed in the Unigraphics as per the requirement given. For that various expressions is to be developed in the expression data base for the model.

8.4.1. Bushing:

In the bushing first requirement is to find the length of the bushing. For that a blank surface is produced at the bottom end at a required dimension from the top. The length is the modifiable parameter and it would be adjusted as per the heat regulation in the bushing. As shown in figure below.

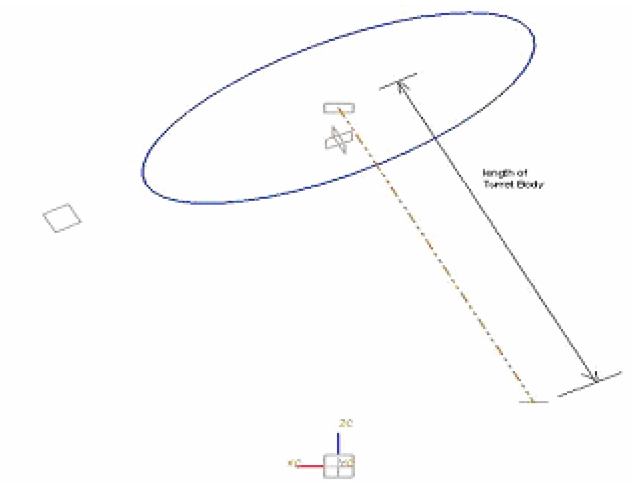


Fig: 8.2. Positioning of Bushing w.r.t Turret

We will set the reference frame at the flange and from there we will made the extrude to the lower face. Now for upper length we know that:

$H = i^2 rt$ -- (1)

and

For a long circular beam due to electric current flux produced will be

$\Theta = BIL$ --- (2)

Here B=magnetic field I= current L=length of conductor

So, from equation (1) and (2) ,once we get the amount of heat dissipation ,than finding the amount of current flowing ,length of bushing can be obtained using

L=BI/Φ-- (3)

So, now that much of bush will be extruded from the flange to the cap height.

Now second problem in this case is that this bush should always be aligned with the turret .so in this case a rotation angle is required to assemble the bush with turret and transformer. This would be clearer from the figure shown below.

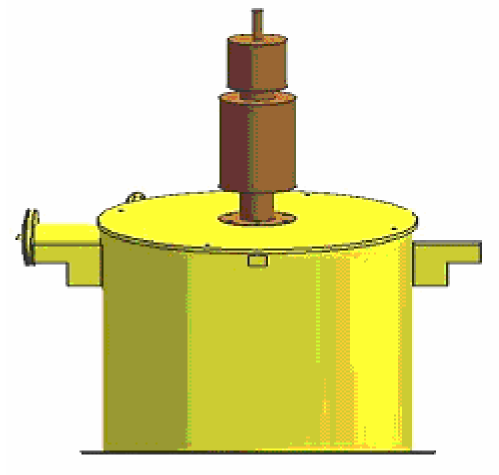


Fig: 8.3. Assembly of bushing and the round turret

8.4.2. Combined angle for assembly of Turret with Bushing:

The major diameter of the ellipse the combined angle is required. As there are two rotation in the round turret. One is about XY plane and the second is about YZ plane. So the rotation can be calculated using the rotation matrix.

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix} * \begin{bmatrix} \cos(\phi) & 0 & -\sin(\phi) \\ 0 & 1 & 0 \\ \sin(\phi) & 0 & \cos(\phi) \end{bmatrix} \text{-------------------------------------- (a)}$$

Let θ be the angle of XY plane and ϕ be angle of YZ plane. Then the combined angle (γ) will be given by:

$\gamma = \arccos(\cos(\theta) * \cos(\gamma))$ --------------------------- (4)

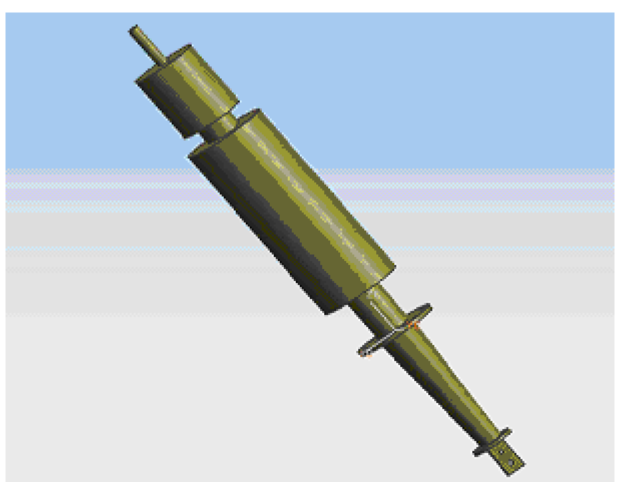


Fig: 8.4. Tilt angle of bushing w.r.t W.C.S

so the above angle comes out to be is the combined angle for the tilting of the Turret

Body and using the relation (4) this assembly and rotation of bushing with turret could be easily governed using Knowledge fusion.

8.5. Integration of the Model with Transformer using Knowledge Fusion:

Now when the modeling is completed the interfacing of the model is done with the user using the knowledge fusion where the placement of the turret assembly is done on the tank body and the expression value which are required to update are updated and the user interface is done. For the user interface (as the user is not required to enter the UG native) the user interface files are made. The figure of that is as follows:

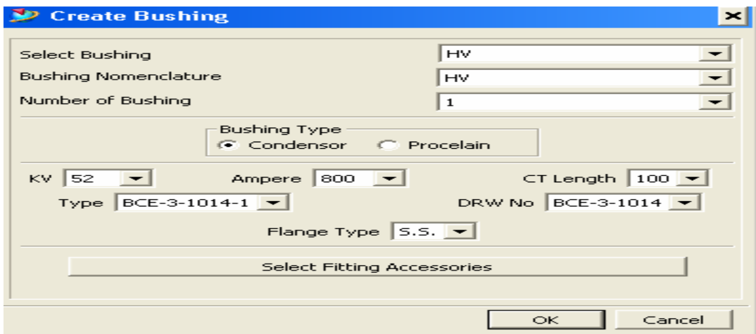


Fig: 8.5. Selection box for selecting Bushing Type

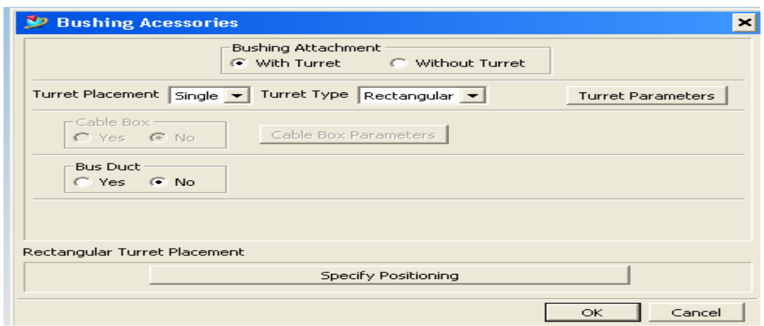


Fig: 8.6. Customized dialog box for selecting accessories

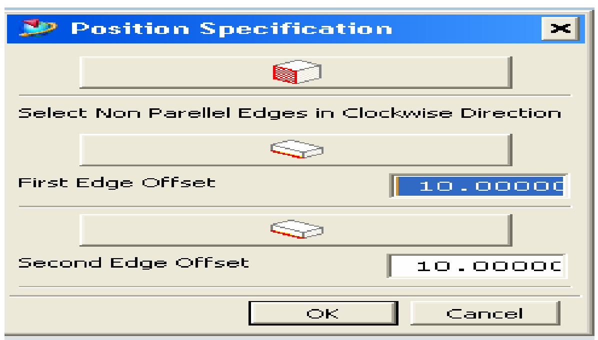


Fig: 8.7. Graphical user interface for Bush positioning

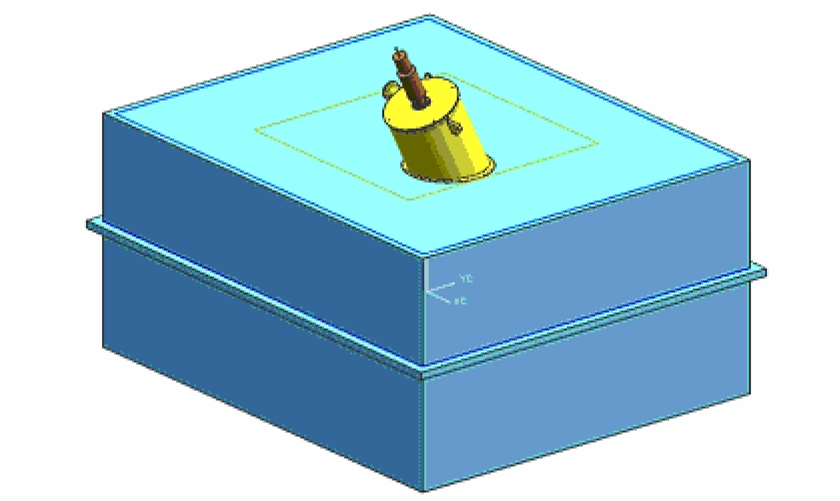


Fig: 8.8. Assembly of Bushing and Turret with Tank body

These files are integrated with the UG using the knowledge fusion. Coding is done in the U-Func for integrating it with the UG. So, after integrating for bushing, user need not make the model. Now user will select only push buttons from dialog box to get the desired model .Bushing with complete assembly of Turret and tank body is shown in figure above.

8.6. Classification or identification of bushing:

The above discussion of creating part families of bushings on the basis of certain parameters directly points to non other than Group Technology. Using GT a part family can be created based on some sort of coding that we had discussed in chapter- 3.

So, the bushing classification based on visual inspection can be done using GT. For that we had used Opitz coding. But finding the similarity ratio directly from a 2D sketch from UG can't be achieved from GT until we get the binary vector of the scanned image. This refers to a sort of feature extraction. This type of conversion is possible in pattern recognition i.e. converting a gray image into a binary vector as we have shown above in chapter-5. For ranking purpose the part with maximum similarity index should be at top. To solve the above problems, AHP will be used as a tool for decision making in the next chapter.

Along with this software usage for classification of bushing will also be discussed.

CHAPTER-9

Software development and results

9.1. Software development and user interface designing [4]

The software application of the methodology developed in Sections 7 and 8 has been implemented in C and C++ in the DOS platform. GT codes and critical design information are structured in C and the data is stored using an object-oriented database system. A user interface has been developed for the X-window environment using C (graphical Interface Library).

Once the user has chosen the Group Technology on the interface screen (Fig: 1), a menu (Fig: 2) with push buttons will appear on which the user will provide the patterns to teach the system. When the buttons of teach will be pushed a scanned data in form of pixels will be shown in teach area which will be stored in database for further similarity resultant.

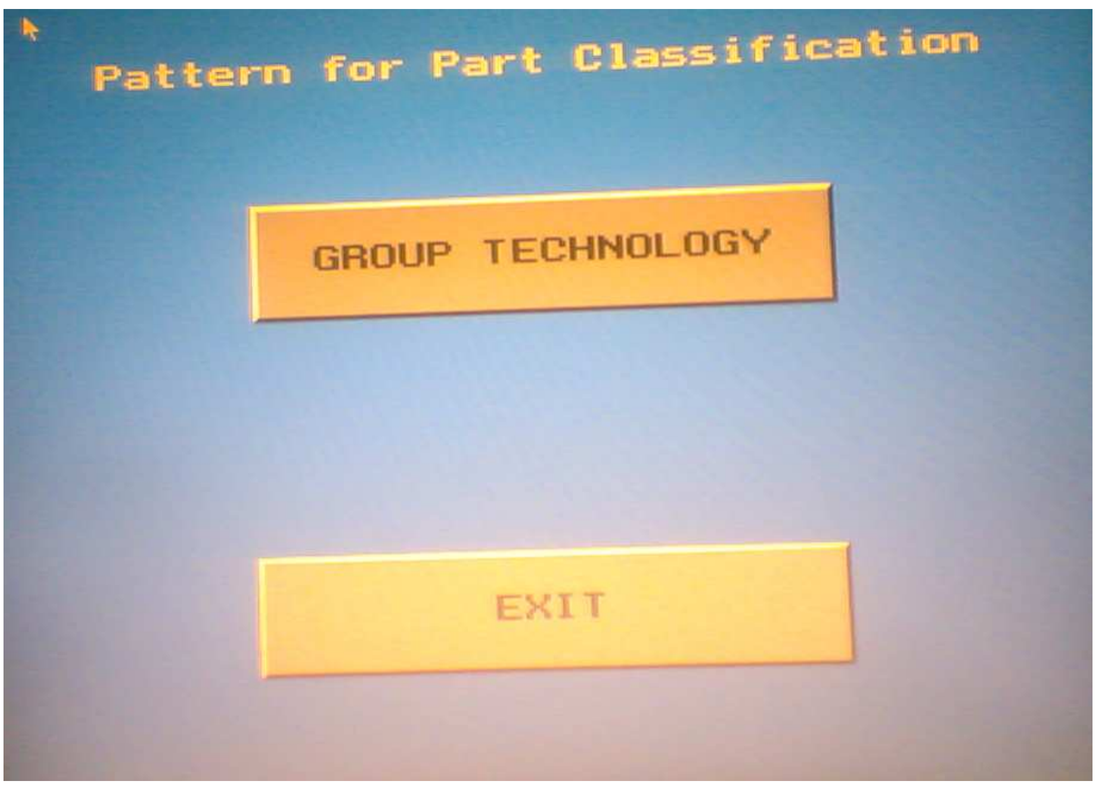


Fig: 9.1. Welcome screen for Pattern Recognition software

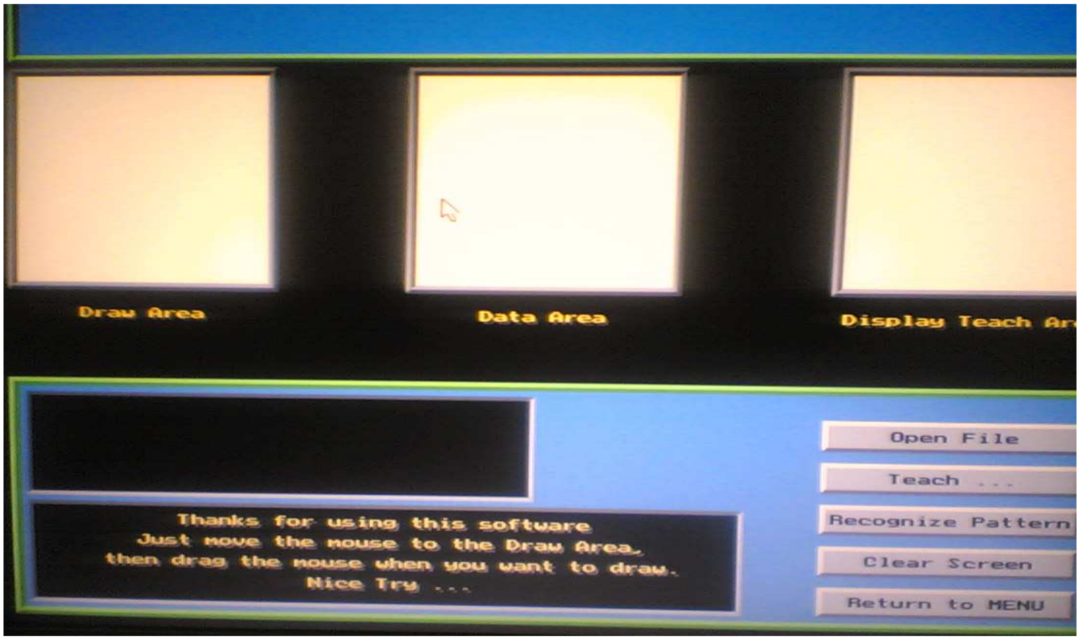


Fig: 9.2. User interface screen with push buttons

The similar set of part can be viewed in Data Area along with information on Similarity percentage ratio , Total no of files and the File no. with which the part in Draw Area is showing the similarity (Fig:3) . For the sorting procedure, we are using here, Bubble sort, so that the part with highest similarity will only be shown in the information window. The user can also get the required help just by hovering the mouse, over any of the push buttons in the help window (as shown above in fig: 2).

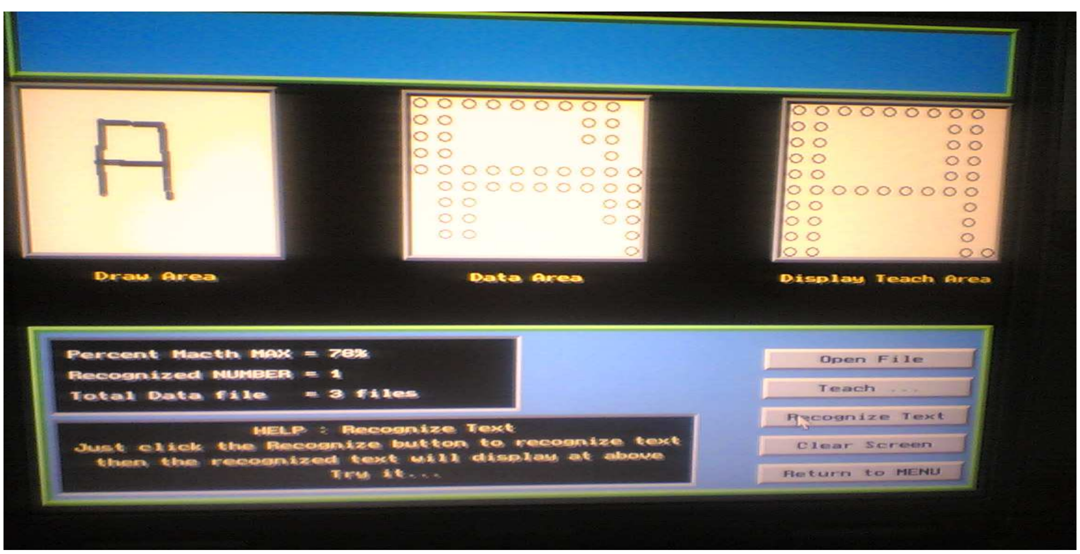


Fig: 9.3. Window showing Similarity index and Data Area

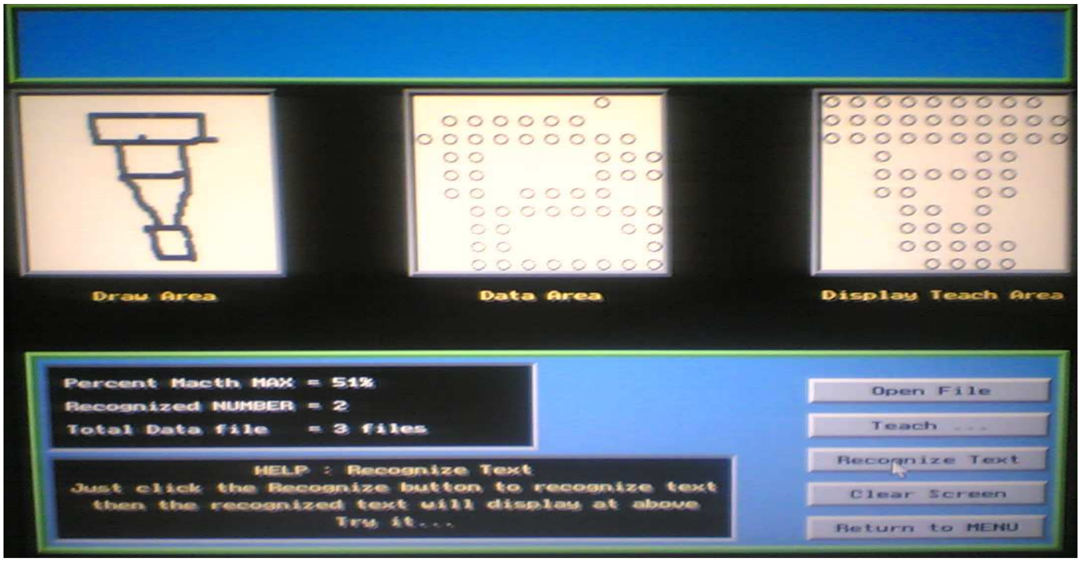


Fig: 9.4. Screen shot showing similarity of Bushing with existing patterns

9.2. Program output & Results:

The methodology was tested by conducting trial runs on both randomly generated databases of parts and industrial parts. The robustness and efficiency of the system was tested by subjecting it to different search and sort intent specifications. Presented in the following sections is the result of whatever going in the intermediate stages to find Similarity Index. A further comparison with Neural net and results on Mat lab will further support the capability of AHP in pattern recognition.

In this simulation only the application of Analytical Hierarchy Process (AHP) with distance function for pattern recognition was performed by taking the user sketch as the candidate part. The following characteristics were chosen for the search intent: (i) main shape from pixel scanning (binary vector); (ii) Area (a feature that works on the principal that whenever there is a pixel would be considered as solid ,while a blank space between x1 and x2 during scanning is treated as a hole or a cut). ; (iii) dimension length i.e. Constant distance (continuous valued, sub type range feature characteristic, like during scanning from x1 to x2 ,the distance covered is (x2-x1)), and (iv) Variable Distance (This would take care of the shapes having variable distance [Ellipse, Circle, Tetrahedron] as compared with shapes having constant distance [square , rectangle , parallelogram etc]).

Another method followed for calculating similarity index is based on visual inspection; in this case we would ask for certain feature based on form code given by opitz. On that basis, software will automatically generate a form code, which will be used further for calculating the SI with respect the existing part families (Fig: 5). than a option is provided for the user to input the desired SI. If the existing part family shows that SI is matching with any of the previous part family than part would be transferred to that family else a new family would be generated for that part.

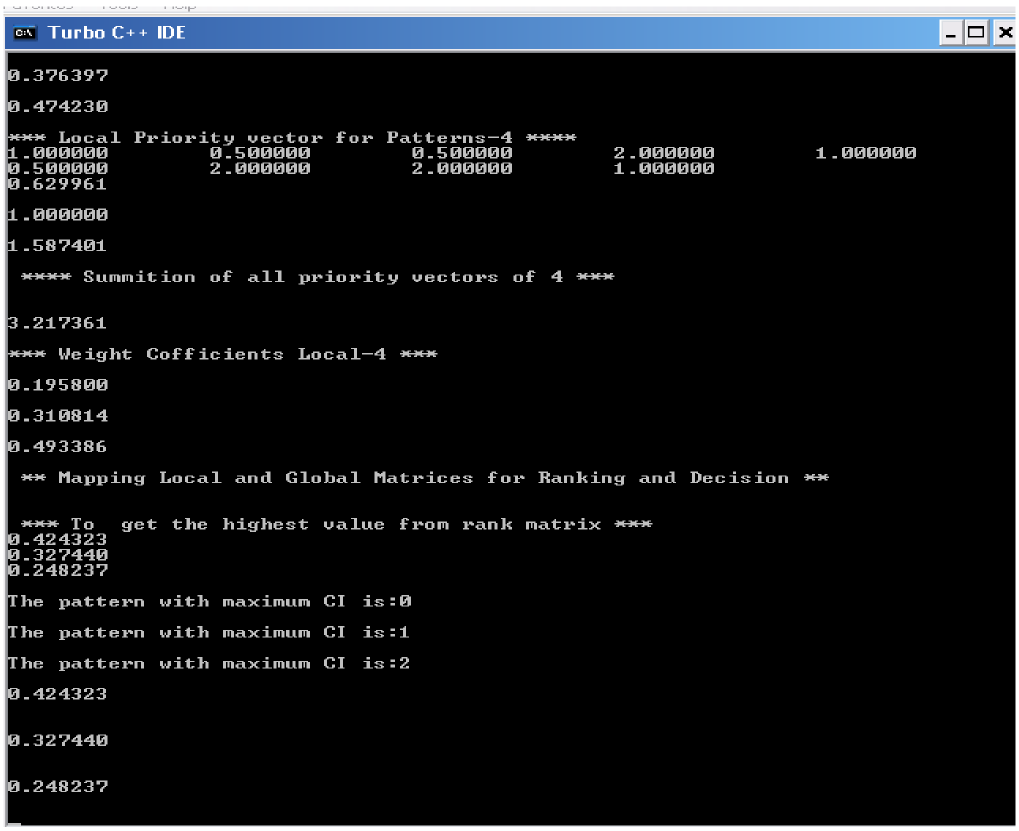


Fig: 9.5. Rank Matrix showing highest similarity Index

9.2.1. Intermediate Results during Calculation of SI

Comparison Of Weighting coefficients for All criterion w.r.t sample A,B and C

	Constant Distance	Variable Distance	Area Scanned	Pixel (Binary Vector)
Sample A	0.652178	0.084232	0.149373	0.1958
Sample B	0.235151	0.333351	0.376397	0.310814
Sample C	0.112671	0.582417	0.47423	0.493386

Table: 9.1 – Comparison of weight coefficient for all criterions

Comparision Of Consistency Index for All criterion w.r.t sample A,B and C

	Constant Distance	Variable Distance	Area Scanned	Pixel (Binary Vector)
Sample A	0.149581	0.024827	0.062263	0.011566
Sample B	0.053933	0.098253	0.156893	0.01836
Sample C	0.025842	0.171663	0.197673	0.029145

Table: 9.2 – Comparison of CI for all DMC at level 2

	Weights
Constant Distance	0.229357
Variable Distance	0.294743
Area Scanned	0.416829
Pixel (Binary Vector)	0.059071

Table: 9.3 – Relative comparison of all DMC at level 2

Final Comparison Matrix for 3 Samples with the input

	A	B	C
Input Index	0.424323	0.32744	0.248237

Table: 9.4 – Final rank table with weighting coefficients

***** Relative Weight for Criterion Research is as Follows *****

0.424323

0.327440

0.248237

In the present problem Part file1 is showing the highest SI for present part.

9.2.2. Results from Mat-Lab:

The Governing equation for calculating all these weights, from Lagrange's method using integrated approach is:

$$L_3 = w^T F w + 2\lambda_3 (e^T w - 1) \quad \text{--- (a)}$$

```
f= [11.04 -2.5 -3.33 -5.20
   -2.5 5.2651 -2.5 -9.11
   -3.33 -2.5 1.4678 -3.33
    -5.20 -9.11 -3.33 -116];

w= [    .2632
         .337
        .4774
         .217    ];

>> s=f*w;
s =      [-0.6549
          -2.0540
          -1.7408
         -31.2005];
>> t=w'*s;       >> t               t = [   -8.4662];

>> e= [ 1
        1
        1
       1];

>> c=e'*w;     >> c=c-1;        >> c=c*2*.7             c = [0.4124];

>> l=c + t;      >> z=peaks (l);          >> z = [weights];

>> Surf (z);               >> colormap (hsv);
```

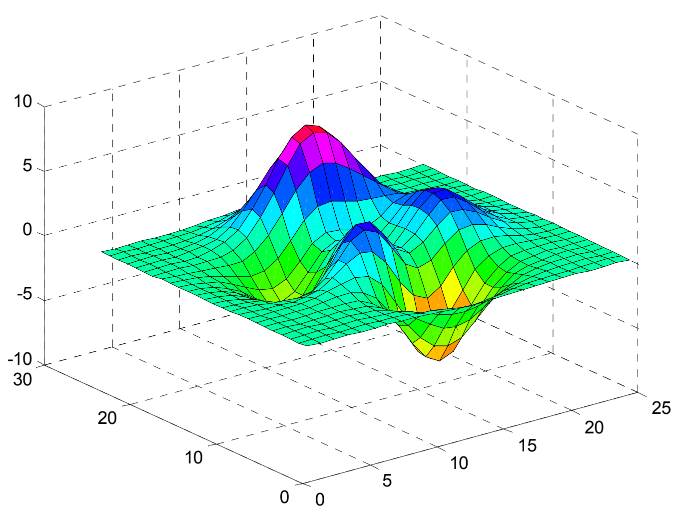
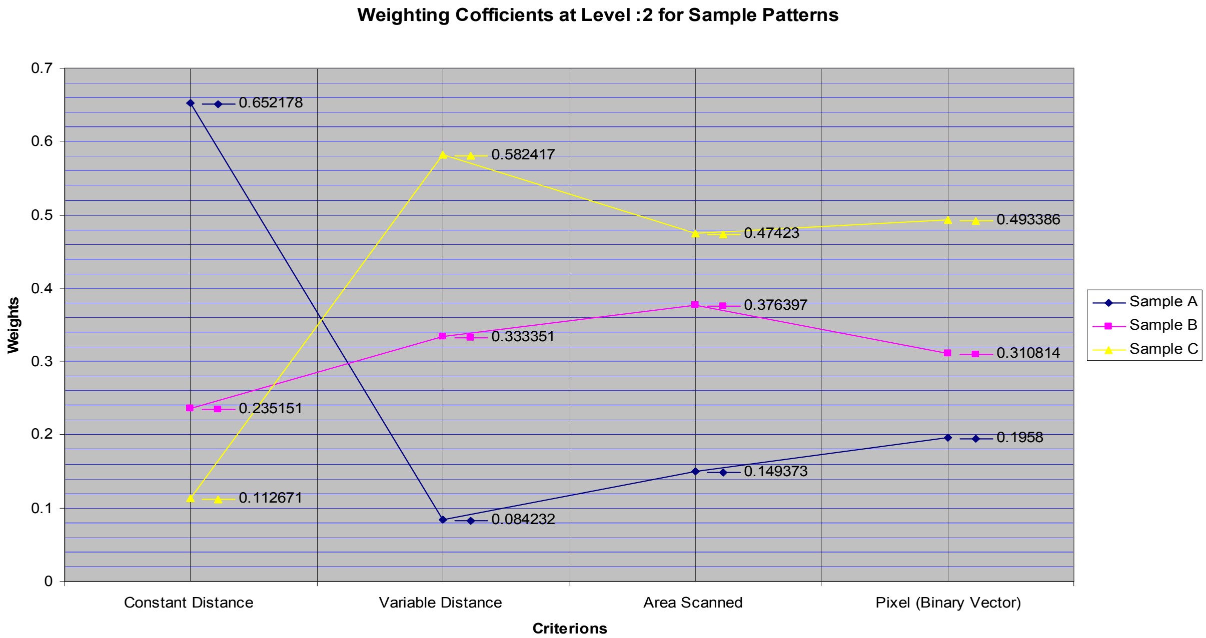
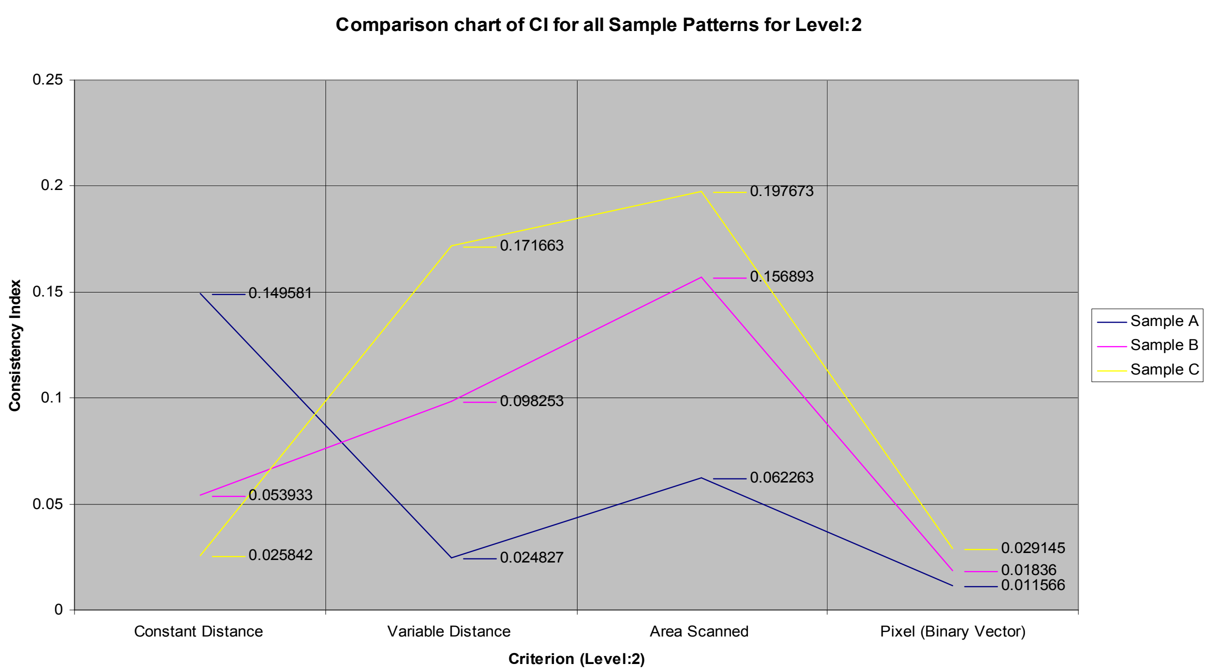


Fig: 9.6. 3D plot for variation of weights at peak values

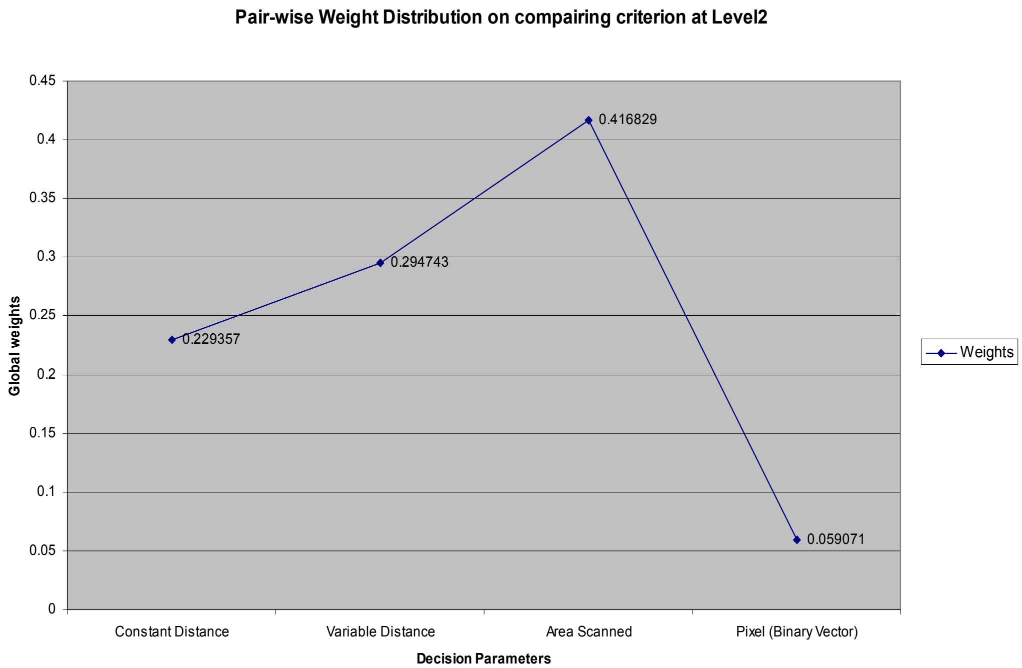
9.2.3. Graphical plots of Intermediate Results:



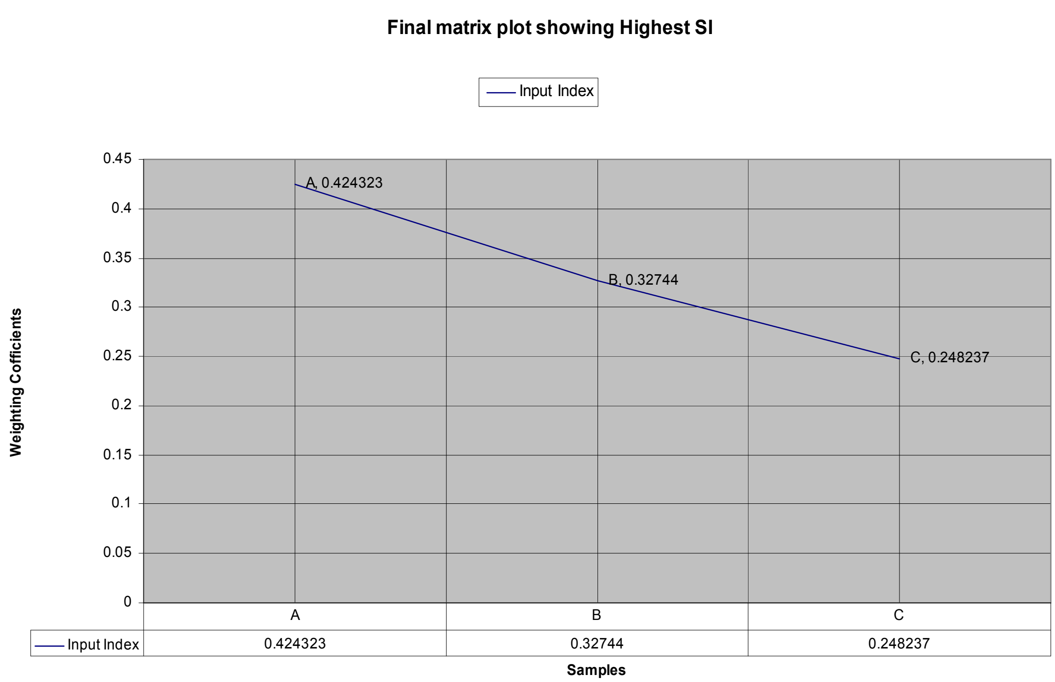
Graph: 9.1. Weighting coefficient for all samples w.r.t DCM



Graph: 9.2. Comparison chart for CI for all samples with respect to DCM at level 2



Graph: 9.3. Pair wise weight distribution on comparing criterion at level 2



Graph: 9.4. Final Rank Matrix plot showing pattern with highest SI

9.2.4. Results from Neural Network:

This Neural Network code is a standard one based on Feed forward Network. It uses Binary sigmoid function as the activation function for both input and output layer. It is most common activation function in ANN's. It can be defined as a strictly increasing function that exhibits smoothness and asymptotes properties. The sigmoid function $g(v)=1/(e^{-av})$ assumes a continuous range of values from 0 to 1.This function is called Logistic function and also gives sigmoid shape.

Here Back propagation network is used for the training purpose. Now the weights for all the 3 samples based on the four criterions in the hidden layer are being used as a weighted samples. These are further used to do the accuracy check on weighted samples from AHP. Table: 5 show the results for all the weighted samples.

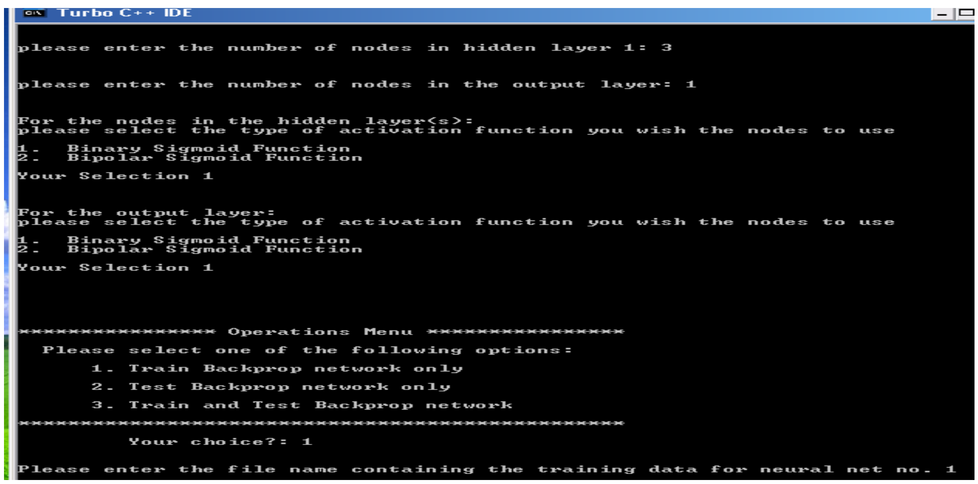


Fig: 9.7. Input given to develop the Feed forward network

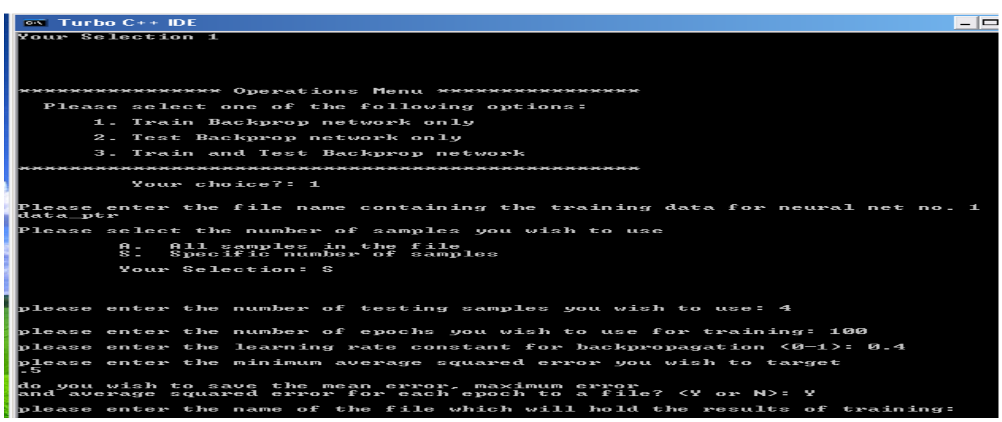
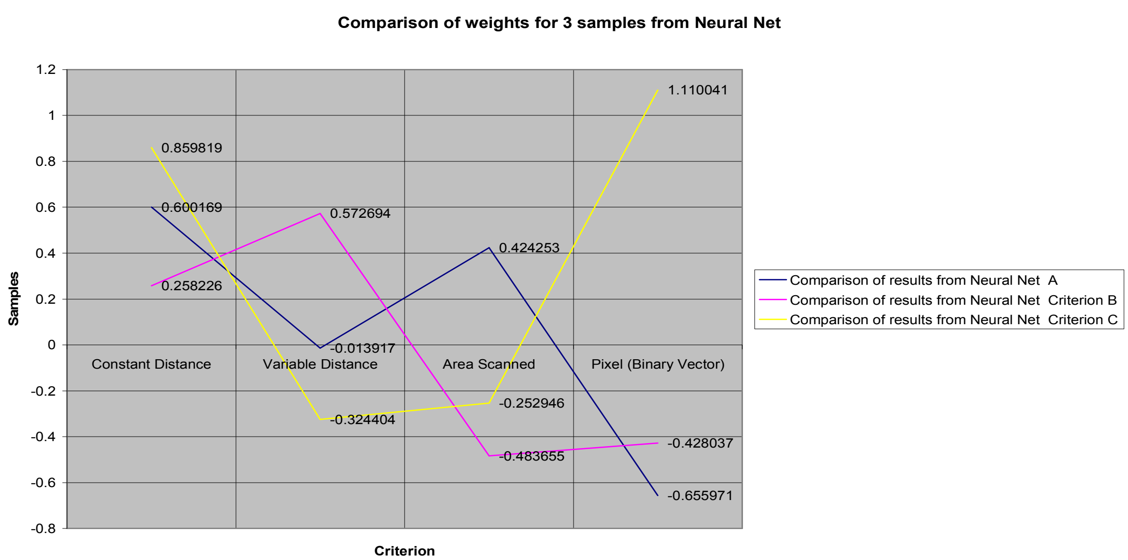


Fig: 9.8. present an overview of learning rate and average squared error

Comparison of results from Neural Net

Samples	A	Criterion B	C
Constant Distance	0.600169	0.258226	0.859819
Variable Distance	-0.013917	0.572694	-0.324404
Area Scanned	0.424253	-0.483655	-0.252946
Pixel (Binary Vector)	-0.655971	-0.428037	1.110041

Table: 9.5. – Final rank table with weighting coefficients using ANN.



Graph: 9.5. Comparing weights for 3 samples on the basis of criterion at Level2

Here in this phase we are comparing the output of weights from neural network code. This would help in deciding the capability of AHP in GT, when compared with respect to Neural Net and Genetic Algorithm. Some screen shots as shown above for Feed forward Neural Network , shows that for the same inputs and for 100 epochs the reliability of both methods are same up to some extent.

The output plotted in the form of line chart can be easily compared with the previous AHP outputs. This shows that though Neural can give good results at 10,000 epochs but for an average application AHP is a much simpler method as compared to neural network. Adding some more benefits AHP can also be used as a classifier in Neural Network or as a member function in Fuzzy logic application. These not only enhances the accuracy but also speed up the processing and decision taking capability of the software.

9.3. Results and Discussion:

It is observed that the application of the system is user friendly and quite self explanatory. The user can operate software without having the detailed knowledge of GT coding schemes or the intricacies of the methodology. If the user is satisfied with the information obtained from the similar parts he has the option of exiting the system.

In the first case, all the characteristics were chosen and the AHP comparison matrix with all the characteristics was used to derive the combination weights.

From above we find that basically it is a three step procedural chain: (1) We calculate the pair-wise weights for all the criterion at level 2. (ii) Second step refers to find weight for sample patterns with respect to each criterion. (iii) Third step composed of finding Consistency Index for all the sample patterns with respect to decision criteria's. Finally CI and weights from global matrix are combined together to get the rank matrix which gives us final rank matrix as resultant.

Table 4, above shows the final rank matrix, in which Part A is showing the maximum similarity index. Corresponding to this graph 4, plotted between samples and weights for criterion is also showing the highest peak at A.

Where as the graph 3 for table 3 is showing a approx Gaussian curve , which provides us with a sort of information that, if we consider the area than weight for this is coming to be maximum. This refers that area is coming out to be a major decision factor in this case. In a similar manner by changing the priority in relative table importance of different criterion can be varied.

Table 1 & 2 mainly shows the graphs for weights and consistency index based on different criterion. This result is found using geometric mean method, but if we find it using integrated weight method than also the results came out to be same .That could be justified by comparing graph 1 & 2 with 3D surface plot from Mat-lab. The peak formation is same as in our 2D graphs. These peaks basically refer to the priority of one criterion over other and finally play a crucial role in determining the overall Similarity index.

The user can explore the various options which will provide insight into the various similarity relationships when comparing parts with respect to specific sets of characteristics.

9.4. Recommendations:

In a dynamic and competitive environment, a manufacturing company must be capable of rapid adjustment in a cost-effective way. One of the key principles to achieve this is to form virtual companies by collaborating with other companies, and by sharing resources, capabilities and information to enable the market demand to be rapidly satisfied. The work discussed in this thesis concerns the need to find similar products to a new product (at the development stage), from the product databases of the collaborating companies. The identification of similar products will provide the designer with their design characteristics and production histories, and reduce the new product's overall lead-time.

Two-step approach has been developed for identifying similar parts. The first step is conversion of gray image to Binary vector that acquires the position of the various pixels for comparing different attributes and desired level of similarity to generate a list of similar parts from partner product databases. The definition of similarity, whole and partial, between the various values of each GT digit has been developed and employed in the similarity search.

The second step of the procedure systematically ranks the similar parts by determining a global similarity measure, based on the search attributes or overall characteristics. Then a combination technique, based on analytical hierarchy process concepts, is employed to combine the similarity measures of the various characteristics into a global measure. This is used to sort the similar part so that the most similar part based on the preference can be immediately identified. During binary conversion the factor added for flexibility also increases the accuracy of the software.

The benefits that can be derived from the above work can be summarized as follows:

(1) The software developed (in C/C++ language) enables the designer to derive the set of similar parts without having to know the details of the entire methodology. This is achieved by the user interface which takes the necessary input from the designer and processes it according to the options specified.

(2) From the sorted list which arranges similar parts in descending order of similarity, the user would able to get the part file showing maximum similarity.

(3) This application gives specific information of the part similarity on the basis of characteristics one has decided to work. If the parameters going to be selected at level: 2 are not up to mark, then it may show ambiguity in results.

(5) From the software development perspective, the modular approach of object-oriented programming will allow future development and revisions to be incorporated with relative ease.

9.5. Concluding Remarks:

The proposed methodology of AHP provides an efficient method to identify similar parts in any organization during designing phase. Software can also provide similarity index on the basis of optiz code for visual inspection. However, some of the assumptions used in the development can be generalized to represent a more realistic scenario: (1) in the binary conversion some of the unrecognized points or pixels are taken in continuous flow, this reduces the similarity index ratio; (2) similarity coefficients can be revised and modified by changing the relative importance of characteristics based on satty relative index but we are taking it as constant.

At present, the methodology developed employs the teaching in first phase for different samples and then proceeds to pattern recognition by sorting in second phase. A possible variation in this methodology is a search option, which derives similarity directly using bubble sorting, after getting SI values from AHP. In this, similarity could be evaluated between the parts, which are treated as objects. This would eliminate the requirement of the GT design processor and any loss of information detail, which could occur in the translation process.

CHAPTER 10

CONCLUSION & FUTURE SCOPE

10.1. Conclusion

The application of KBE through UG for design and automation of the transformer has reduced the overall cost as well as the lead-time for developing a new transformer as per the requirements of the customer. In statistical terms the ratio is 15% and 45% respectively. This concludes that Knowledge based systems are the future of the industry. One has to be with the technology to survive in the present age of cut throat competition.

This not only reduces the time and effort but also enhances the overall working process. Secondly the part recognition feature using opitz coding also enhances the search capabilities i.e. if any new part is also coming in the library than that profile could be matched with the previous one and solution could be sorted out in very less time. Along with KBE the AHP technology for pattern recognition has also open the new gates to develop the fully automated GT cells in a much simpler manner. The application of pattern recognition with AHP has definitely solved the problem of modeling the large no of parts with similar features. Along with this the application of integrated weight to AHP has increased the efficiency of the software by 2 folds.

From above results it can be deduced, that “Pattern recognition with GT” has not only solved the problem of part classification but also provides a rapid sorting method using which user can sort the part as per his preferences on the basis of Similarity index. This method is much simpler as compared to the existing neural net approach. One more advantage of using AHP is that it can also be used as a classifier function with Neural net or as a member function with Fuzzy to stimulate the decision-making capabilities of these systems. This shows that earlier systems with Neural or fuzzy based can also be integrated just by modifying the classifier or member function.

So, the onus lies on us to brave the wind and perpetuate the technology in a way that it could revolutionize the industry by benefiting the owner as well as the customer.

10.2. Future Scope

KBE is a subset of Knowledge Based Systems (KBS), which is a spin off from artificial intelligence (AI). KBS is often referred to as expert system" because they intend to capture expert knowledge and sometimes also generate creative solutions. So there is enhanced scope of KBE to append the KBE up to KBS, so that it would be used as either an expert system or AI in product development process.

On the other hand the AHP process can be further integrated with fuzzy system to get a higher accuracy. This will also increase and enhance the decision taking capability of the system. Presently what we are getting as output is the Similarity Index and the part number, while this could be extended to provide the list of all those parts to user which shows a SI greater than as input by the user. This would provide an option to user, to sort the parts according to the features he required.

Further this pattern recognition can be modified to 3D; in that case the software could be integrated even for automated assembly sequencing. The statement can be justified because, as we know that for automated assembly sequencing we require an ultra capable decision classifier, which could handle, the combinatorial explosion generated during assembly sequencing. In the present work we have used AHP, a decision classifier for pattern recognition, so no other classifier is required for sequencing. Hence we can say that just by modifying the software for 3D Pattern Recognition, we can achieve the most typical task of present industry that is "Automated Assembly sequencing & planning".

Chapter -11

REFRENCES

1. Andrew Godbehere, Engineering Programme Manager Airbus UK, Page 29, December 2004.
2. Andrew Ellis, Chairman, Central Canada Regional Users Group (CCRUG), Volume 5, Issue 3, Dec 2002.
3. Anil.K.Jain , Feature Extraction Methods For Character Recognition- A Survey , Pattern Recognition, Vol. 29, No. 4, pp. 641-662, 1996.
4. Anna M. Cox, Jeffrey Alwang, Thomas G. Johnson, "Local Preferences for Economic Development Outcomes: An Application of the Analytical Hierarchy Procedure", Community Policy Analysis Center, September 1999 , (www.nercrd.psu.edu/Industry_Targeting/ExtensionPapersandSlides/IndCluster Ext.JohnsonApproach)
5. Alessio Ishizaka, Advantages of clusters and pivots in AHP, University of Basel, Switzerland, (http://www.wwz.unibas.ch/wi/members/ishizaka/MUDSM04.pdf).
6. A.Delchambre, Computer Aided assembly planning, Chapman & Hall, New York, 1990.
7. Boart, P., Sandberg, M., Nergård, H., and Isaksson, O. A .knowledge enabled engineering approach for conceptual design of life cycle properties. Submitted to Journal of Computing and Information Science in Engineering.
8. Billo, Richard E.; Bidanda, Bopaya, "Representing group technology classification and coding techniques with object oriented modeling principles." IIE Transactions , August 1995 issue.
9. Craig B. Chapman, Martyn Pinfold, "The application of a knowledge based engineering approach to the rapid design and analysis of an automotive structure", advances in engineering softwares 32 (2001), pp. 903-912.
10. Dirk Cornelis Uys conradie, "The use of software system to implement case based reasoning enabled intelligent component for production briefing and design", October 2000.
11. Daniel L. Schmoldt and David L. Peterson, "Analytical Group Decision Making in Natural Resources: Methodology and Application", Journal of Forest Sciences, 2000 – (science.nature.nps.gov).

12. Enrico Franconi, Description Logics for Conceptual Design, Information Access , and Ontology Integration: Research Trends, http://www.inf.unibz.it// franconi
13. "Grip Fundamentals student guide", January 2000, MT13010-version 16.0
14. Gianni Caligiana, Alfredo Liverani, Stefano Pippa, Modeling , Design and Analysis of a Testing Rig For Composite Materials , DIEM – Facoltà di Ingegneria , Viale Risorgimento , 2/ 40136 Bologna Italy.
15. Dr.H.M.Chan, "Group Technology as a Manufacturing Philosophy", Ph.D. Thesis, University of Aston, Birmingham, U.K. 1981.
16. Iyer, Shekhar; Nagi, Rakesh, "Automated retrieval and ranking of similar parts in agile manufacturing", IIE Transactions: Oct, 1997 issue.
17. John S. Gero, An overview of knowledge engineering and its relevance to CAAD", CAAD Futures Digital Proceedings 1986.
18. JM & Co/AJRA, the Basics of Expert (Knowledge Based) Systems, IEEE Proceedings 1997.
19. János Fülöp, "Introduction to Decision Making Methods", Computer and Automation Institute, Hungarian Academy of Sciences (http://www.evergreen.edu/bdei/documents/decisionmakingmethods.pdf.)
20. Jian Ma, Zhi-Ping Fan and Li-Hua Huang "A Subjective and Objective Approach to Determine Attribute Weights" , University of Hong Kong, International Journal of operation research and decision making , Volume-3, September 2005.
21. Kevin. L. Moore , "Iterative Learning Control" , Proceedings of the 34th Conference on Decision and Control , New Orleans , LA , July 1995
22. L´aszl´o Felf¨oldi, Andr´as Kocsor, "AHP-Based Classifier Combination", Journal ASME, Volume September 2005, (www.inf.u-szeged.hu/kocsor/publications/ Papers/2004/Conf-2004-ICEIS-FL/Web/FKo04.pdf)
23. Marcus Sandburg, "Knowledge Based Engineering – in product development", Division of Computer Aided Design, 2003.
24. Marcus Sandberg, "Knowledge enabled engineering design tools for manufacturability evaluation of jet engine components", Division of Computer Aided Design, SE-971 87 Luleå, Sweden, March 2005
25. MITCHELL M. TSENG and JIANXIN JIAO, "Case-based evolutionary design for mass customization", Computers and Industrial Engineering Vol.33, No.1-2, 1997, pp.319-324.

26. Michael Wilson, Knowledge Engineering, SOFSEM '93 - Hrdonov, Sumava, Czech Republic, 21.11. - 3.12.1993
27. M. Zimmermann, R. Bronsart, "Application of knowledge-based engineering methods for standardization and quality assurance in ship structural design", September 1998.
28. "NX Digital Simulation, Experimentation is the key to innovation", www.ugs.com
29. Peter wood , university of Queensland , (ftp//192.168.256)
30. ppt , Training Material UI styler , Tata Consultancy services
31. Sandberg, M., Åström, P., Larsson, T. and Näström, M. A design tool integrating CAD and virtual manufacturing for distortion assessment. Accepted for presentation at International conference of Engineering Design, Melbourne, August 15-18 2005.
32. Dr. Stephan Zelewski, Malte L. Peters, "A heuristic algorithm to improve the consistency of judgments in the Analytical Hierarchy Process (AHP)", Institute for Production and Industrial Information management ,University Duisburg-Essen, (www.pim.uni-due.de/fileadmin/Publikationen/ Arbeitsbericht_nr_18_AHP_Heuristik.pdf)
33. "Transformers Second Edition", Bharat Heavy Electricals Limited, Bhopal, pp.577-580, pp.562-576 TMH publication, 2003.
34. Dr. T.C. Chang, "GROUP TECHNOLOGY", School of Industrial Engineering, Purdue University, October – 1999.
35. Thomas L. Saaty, Analytic Hierarchy Process, 1990 (product.snu.ac.kr/prodproc/ch07a-AHP).
36. Thomas L. Saaty, "How to make a decision: The Analytic Hierarchy Process", European Journal of Operational Research, 48 (1990) 9-26, North-Holland.
37. "Unigraphics NX technical Briefing, For Total Product Engineering", August 2002.
38. "Unigraphics NX technical Briefing, For Total Product Engineering", August 2002.
39. "UG/open API student guide", October 1999, MT13010-version 16.0.
40. V. Kalyanapasupathy, E. Lin1, I. Minis, "Group Technology Code Generation over the Internet", University of Maryland , http://www.isr.umd.edu/Labs/CIM/ profiles/lin/docs/gt/asme97-1c.rtf
41. Let Us C, by Yashwant Kanitkar, BPB publications, India.

References from NET

42. http://www.mcadonline.com
43. http://www.eil.utoronto.ca/kbd/ditl/index.html
44. http://www.sbg.bio.ic.ac.uk/ilp/bbsrc/proposal/node3.html
45. http://www.itcon.org/2002/2
46. http://www.findarticles.com/p/articles/mi_qa3685/is_199501/ai_n8729902
47. http://www.ielm.ust.hk/dfaculty/ajay/courses/ieem513/GT/GT.html
48. http://www.allbusiness.com/periodicals/article/523111-3.html
49. http://idmru.aston.ac.uk/camac
50. http://wwwlib.umi.com/dissertations/preview_page/8826752/14
51. http// 204.96.58//Alberta university
52. http:// www.allbusiness.com/periodicals/article/652844-7.html
53. www.library.unsw.edu.au/thesis/adt-NUN/uploads/approved/adt-NUN20040818.094936/ public//04chapter3.pdf