Rule Based Setup and Fixture Planning for Prismatic Parts on 3-Axis and 4-Axis Milling Machines

A thesis presented to
the faculty of
the Russ College of Engineering and Technology of Ohio University

In partial fulfillment
of the requirements for the degree
Master of Science

Mayur M. Wakhare
August 2016

© 2016 Mayur Wakhare. All Rights Reserved.
This thesis titled
Rule Based Setup and Fixture Planning for Prismatic Parts on 3-Axis and 4-Axis Milling Machines

By
MAYUR M. WAKHARE

has been approved for
the Department of Industrial and Systems Engineering
and the Russ College of Engineering and Technology of Ohio University by

Dušan N. Šormaz
Associate Professor of Industrial and Systems Engineering

Dennis Irwin
Dean, Russ College of Engineering and Technology
ABSTRACT

WAKHARE, MAYUR M., M.S., August 2016, Industrial and Systems Engineering

Rule Based Setup and Fixture Planning for Prismatic Parts on 3-Axis and 4-Axis Milling Machines

Director of Thesis: Dušan N. Šormaz

This thesis developed an architecture for rule based setup planning for prismatic parts on 3-axis and 4-axis milling machines and implemented its prototype. The rule based setup planning system ‘RB-SPS’ generates the minimum number of setups for a given prismatic parts on 3 and 4-axis milling machines using heuristic procedure. The steps involved in the setup planning are: Feature and Process clustering, Setup formation, Operation sequencing and Setup sequencing. Those steps are accomplished by successive execution of rules based on part geometry, its features and previously selected machining processes. The prototype has been integrated with the IMPlanner system and GUI has been implemented for monitoring of steps for setups generation. The GUI provides the necessary functionality for the user to navigate the system.

The developed RB-SPS system has been tested on real mechanical prismatic parts and successful results are achieved.
ACKNOWLEDGMENTS

I am deeply thankful to all the people who helped me, directly or indirectly, during this process of completing my Master’s thesis.

I would like to express my heartfelt gratitude to my thesis advisor Dr. Dušan Šormaz for his continuous guidance and encouragement. His inclination for critical thinking, hard work, passion and his helpful attitude are some of the qualities that I hope to inculcate in myself, as they will help me throughout my life. It has been an immense pleasure working with him.

It has been a pleasure to pursue graduate studies in ISE department of OU which provided me with some great exposure to enhance my skills on both personal and professional platform.

I would like to thank my committee members Dr. Kuruppalil, Dr. Huq and Dr. Yuan for their suggestions and frank discussions I needed for the thesis.

I would like to thank Manish, Shantanu, Tejas, Pratik, Ajinkya and all my other friends and colleagues who helped me during my graduate studies and provided valuable advice.

I thank my family members for their support, especially my uncle Mr. Vikas Satav for the timely financial aid.

This acknowledgement would not be complete without thanking two of the most important people in my life – my Father, Madhukar and my Mother, Bharati. I thank them for all their support and encouragement. It helped me a lot during my life as a master’s student at Ohio University.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>3</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td>4</td>
</tr>
<tr>
<td>List of Tables</td>
<td>8</td>
</tr>
<tr>
<td>List of Figures</td>
<td>9</td>
</tr>
<tr>
<td>Nomenclature</td>
<td>14</td>
</tr>
<tr>
<td>1. Introduction</td>
<td>15</td>
</tr>
<tr>
<td>2. Literature Review</td>
<td>18</td>
</tr>
<tr>
<td>2.1 Multiple-Constraints</td>
<td>18</td>
</tr>
<tr>
<td>2.2 Fixture Driven Setup Planning</td>
<td>21</td>
</tr>
<tr>
<td>2.3 Tolerance Analysis</td>
<td>23</td>
</tr>
<tr>
<td>3. Problem Statement</td>
<td>25</td>
</tr>
<tr>
<td>4. Objective</td>
<td>27</td>
</tr>
<tr>
<td>5. Methodology</td>
<td>29</td>
</tr>
<tr>
<td>5.1 IMPlanner System</td>
<td>29</td>
</tr>
<tr>
<td>5.2 Setup and Fixture Planning Module</td>
<td>31</td>
</tr>
<tr>
<td>6.1 Identification of Tool Approach Directions (TAD) for Feature Clustering</td>
<td>37</td>
</tr>
</tbody>
</table>
6.2 Feature and Process Clustering by TAD and GD&T ............................................. 43
6.3 Datum Precedence (Tolerance Precedence Network - TPN)................................. 49
6.4 Setup Sequencing.................................................................................................... 54
   6.4.1 Three Feature Conflict ..................................................................................... 55
   6.4.2 Four Features Conflict ..................................................................................... 56
6.5 Operation Sequencing within the Setups ............................................................... 63
6.6 Setups Precedence and Ordering .......................................................................... 66
   6.6.1 Setup Precedence ............................................................................................. 66
   6.6.2 Setup Ordering .................................................................................................. 68
7. Setup Planning for 4-Axis Milling Machine ............................................................. 72
   7.1 Generation of 4-Axis Setups ................................................................................ 75
   7.2 Selection of 4-Axis Setups ................................................................................... 81
   7.3 Four-Axis Setups Sequencing Based on FPN and TPN ......................................... 82
   7.4 Operation Sequencing within the 4-Axis Setups .................................................. 83
   7.5 Four-Axis Setups Precedence and Ordering .......................................................... 86
8. Fixture Planning ....................................................................................................... 87
   8.1 Methodology ......................................................................................................... 87
9. Implementation ......................................................................................................... 90
   9.1 Rule Based-Setup Planning System (RB-SPS)....................................................... 90
9.2 Integration of IMPlanner and RB-SPS ................................................................. 97

9.2.1 Integration Panel ............................................................................................ 97

9.2.2 Tolerance Definition Panel ........................................................................... 100

9.2.3 Setup Planning Panel .................................................................................... 102

10. Case Studies .................................................................................................... 110

10.1 Slider .............................................................................................................. 110

10.2 ANC101 ......................................................................................................... 121

10.3 Jaw-Stationary ................................................................................................. 134

11. Conclusions and Future Work ........................................................................ 140

11.1 Conclusion ..................................................................................................... 140

11.2 Limitations ..................................................................................................... 140

11.3 Future Work ................................................................................................... 141

References ............................................................................................................ 143
LIST OF TABLES

Table 1 Processes and TADs for features in example ...................................................... 46


LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Architecture of IMPlanner system</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>Architecture of setup and fixture planning system</td>
<td>32</td>
</tr>
<tr>
<td>3</td>
<td>Vertical 3-axis milling machine [25]</td>
<td>36</td>
</tr>
<tr>
<td>4</td>
<td>Tool approach directions for different features</td>
<td>39</td>
</tr>
<tr>
<td>5</td>
<td>TAD rule for blind-hole feature</td>
<td>40</td>
</tr>
<tr>
<td>6</td>
<td>TAD rule for through-hole feature</td>
<td>41</td>
</tr>
<tr>
<td>7</td>
<td>TAD when it exists</td>
<td>42</td>
</tr>
<tr>
<td>8</td>
<td>Facts for TAD rule of blind-hole feature</td>
<td>43</td>
</tr>
<tr>
<td>9</td>
<td>TADs for blind and though hole feature</td>
<td>43</td>
</tr>
<tr>
<td>10</td>
<td>Tolerance relationship between features for example</td>
<td>44</td>
</tr>
<tr>
<td>11</td>
<td>Rule to cluster features that have tolerance relationship between them</td>
<td>47</td>
</tr>
<tr>
<td>12</td>
<td>Clustered features</td>
<td>48</td>
</tr>
<tr>
<td>13</td>
<td>Setup formation based on tolerance relation and TAD</td>
<td>49</td>
</tr>
<tr>
<td>14</td>
<td>Rule for datum precedence</td>
<td>50</td>
</tr>
<tr>
<td>15</td>
<td>Datum precedence (TPN)</td>
<td>50</td>
</tr>
<tr>
<td>16</td>
<td>Slider</td>
<td>51</td>
</tr>
<tr>
<td>17</td>
<td>Datum precedence (TPN) for Slider</td>
<td>52</td>
</tr>
<tr>
<td>18</td>
<td>Setup formation of Slider part</td>
<td>53</td>
</tr>
<tr>
<td>19</td>
<td>Setup formation of Slider part</td>
<td>54</td>
</tr>
<tr>
<td>20</td>
<td>Three features conflicts</td>
<td>56</td>
</tr>
</tbody>
</table>
Figure 21 Four features conflict ................................................................. 57
Figure 22 Setup sequencing rule ................................................................. 58
Figure 23 Setup conflicts in augmented FPN .............................................. 59
Figure 24 Setup conflict resolution ............................................................. 60
Figure 25 New setups after setup conflict resolution .................................. 62
Figure 26 Operation sequencing within setup when no previous features ..... 63
Figure 27 Operation sequencing within setup for Slider ............................ 65
Figure 28 Rule for setup precedence ......................................................... 66
Figure 29 Three-axis setups precedence for Slider .................................... 68
Figure 30 Rule to order setups for 3-axis mill ............................................ 69
Figure 31 Ordered setups for Slider ........................................................... 70
Figure 32 Horizontal 4-axis milling machine [27] ...................................... 72
Figure 33 Other types of 4-axis milling machines [28] ............................... 74
Figure 34 Vertical 4-axis milling machine [28] .......................................... 75
Figure 35 Making 4-axis setups ................................................................. 76
Figure 36 Rule for making 4-axis setups .................................................... 77
Figure 37 Add features and processes from remaining 3-axis to existing 4-axis setups .................................................. 78
Figure 38 Generated 4-axis setup of vector (0 0 1) for Slider .................... 79
Figure 39 Generated 4-axis setup of vector (1 0 0) for Slider ..................... 80
Figure 40 Generated 4-axis setup of vector (0 -1 0) for Slider .................... 81
Figure 41 Finalizing 4-axis setups from total possible 4-axis setups .......... 81
Figure 42 Finalized 4-axis setups without setup conflicts ......................... 83
Figure 43 Operation sequencing within 4-axis setup (vector - \((0 0 1)\)) for Slider.............. 84
Figure 44 Operation sequencing within 4-axis setup (vector - \((-1 0 0)\)) for Slider .......... 85
Figure 45 Four-axis setup precedence for Slider.............................................................. 86
Figure 46 The 3-2-1 locating principle for prismatic part [29].............................................. 87
Figure 47 Choose a part surface for 3-locating points....................................................... 88
Figure 48 An architecture of rule based system................................................................. 91
Figure 49 Feature template for setup planning .................................................................. 92
Figure 50 TAD template for setup planning...................................................................... 93
Figure 51 Process template for setup planning.................................................................. 93
Figure 52 Tolerance template for setup planning .............................................................. 94
Figure 53 Three-axis setup template for setup planning ................................................... 94
Figure 54 Mapped features of Slider and generated processes in Integration Panel ........ 98
Figure 55 XML file saved by Integration Panel with alternate processes for Slider........... 99
Figure 56 Tolerance Definition Panel............................................................................. 101
Figure 57 XML file saved by Tolerance Definition Panel that shows Slider tolerances 102
Figure 58 Setup Planning Panel when 4-axis machine is available.................................... 103
Figure 59 Setup Planning Panel when 4-axis machine is not available............................ 103
Figure 60 Datum facts creation rule for Slider ................................................................. 104
Figure 61 A pop up menu when right clicked on any process after the setup planning . 107
Figure 62 Visualization of generated 3-axis setups of Slider in Setup Planning Panel .. 108
Figure 63 Visualization of generated 4-axis setups for Slider in Setup Planning Panel. 109
Figure 64 Slider.............................................................................................................. 110
Figure 65 2D drawing of Slider ................................................................. 111
Figure 66 Alternate process generation for Slider in Integration Panel ....................... 111
Figure 67 XML file saved by Integration Panel with alternate processes for Slider ..... 112
Figure 68 Adding tolerances for Slider features ....................................................... 113
Figure 69 XML file saved by Tolerance Definition Panel for Slider ........................... 114
Figure 70 Generated 3-axis setups for Slider .......................................................... 115
Figure 71 Generated 3-axis setups and TADs for Slider shown geometrically ............. 116
Figure 72 Generated 4-axis setups for Slider ............................................................ 117
Figure 73 TPN for Slider ...................................................................................... 117
Figure 74 Feature precedence network of Slider ....................................................... 118
Figure 75 Augmented FPN (with 4-axis setups) of Slider ........................................ 119
Figure 76 Generated 3-axis setups precedence of Slider .......................................... 119
Figure 77 Generated 4-axis setups precedence of Slider .......................................... 120
Figure 78 Generated 4-axis setups and TADs for Slider shown geometrically .......... 121
Figure 79 ANC101 ............................................................................................. 121
Figure 80 2D drawing of ANC101- side view ............................................................ 122
Figure 81 2D drawing of ANC101- top view ............................................................. 123
Figure 82 Mapped features of ANC101 and generated processes in Integration Panel . 124
Figure 83 XML file saved by Integration Panel with alternate processes for ANC101 . 125
Figure 84 Adding tolerances for ANC101 features .................................................... 126
Figure 85 Display of generated 3-axis setups of ANC101 in Setup Planning Panel ...... 127
Figure 86 Generated TPN for ANC101 part ............................................................. 128
Figure 87 Generated augmented FPN with color-coded 3-axis setups ....................... 129
Figure 88 Three-axis setups precedence graph of ANC101 ................................. 129
Figure 89 Generated 3-axis setups and TADs for ANC101 shown geometrically ..... 130
Figure 90 Display of generated 4-axis setups of ANC101 in Setup Planning Panel ... 131
Figure 91 Augmented FPN of ANC101 with generated 4-axis setups .................... 132
Figure 92 Four-axis setup precedence for ANC101 ............................................. 133
Figure 93 Generated 4-axis setups and TADs for ANC101 shown geometrically ...... 133
Figure 94 Jaw-Stationary .................................................................................... 134
Figure 95 2D drawing of Jaw-Stationary ............................................................. 135
Figure 96 Alternate process generation for Jaw-Stationary part in Integration Panel ... 135
Figure 97 Generated 3-axis setup for Jaw-Stationary .......................................... 136
Figure 98 TPN of Jaw-Stationary part ................................................................. 137
Figure 99 Augmented FPN for Jaw-Stationary part ............................................ 137
Figure 100 Generated 3-axis setups and TADs for Jaw-Stationary shown geometrically ................................................................. 138
NOMENCLATURE

- **CAD (Computer Aided Design)** – The use of computer in both representation and analysis steps in design.

- **CAM (Computer Aided Manufacturing)** – The use of computer software to control machine tools and processes in the manufacturing of workpiece.

- **CAPP (Computer Aided Process Planning)** – The function which uses the computer to assist the work of process planners.

- **IMPlanner (Intelligent Manufacturing Planner) System** – The CAPP prototype developed in Ohio University.

- **RB-SPS (Rule Based – Setup Planning System)** – It is the setup planning module developed in this thesis research.

- **Setup** – A setup is the way in which the part is oriented and fixtured in one particular position during machining.

- **TAD (Tool Approach Direction)** – The machining process axis is defined as the vector of the tool over the workpiece during machining of the features.

- **GD&T (Geometric Dimensioning and Tolerancing)** – The system that specifies tolerances based on how part is to function.

- **TPN (Tolerance Precedence Network)** – It states that which datum (feature) should be machined prior to which feature.

- **FPN (Feature Precedence Network)** – Shows that which feature should be made prior to which feature considering geometric and technological constraints.

- **Augmented FPN** – It is the combination of TPN and FPN.
1. INTRODUCTION

Process planning is a critical bridge between design and manufacturing [1]. Process planning takes the design information of a part, such as dimensions, material, tolerances, etc. and converts it into orderly manufacturing steps to produce a finished product from the raw stock. It can be defined as an activity to get the detailed instructions to manufacture the part at low cost and high quality with available resources.

Traditional process planning is a time consuming process and tedious too. Traditional process planning involves the generation of manufacturing instructions based on experience and knowledge of manufacturing experts [1] [2]. Today, to enhance the ability of enterprises to quickly respond to the dynamic changes as well as in mass production computer aided process planning (CAPP) concept was developed[3].

Computer aided process planning is a smooth integration between computer aided design (CAD) and computer aided manufacturing (CAM). Currently, in the competitive market, meeting customer’s variety of demands as well as continues redesigning of the product in a very short time is a big challenge. CAPP significantly reduces the actual time right from design to manufacture the product[3].

Setup planning is an integral part of the process planning. Setup is a way to orient the part and secure it with the fixtures in one particular position on the machine bed during machining[4]. A group of manufacturing features are machined in a setup without changing the part orientation. When the orientation needs to change then it is considered as a new setup. Setup planning consists of following tasks[4]:


1. Recognition of machining features and generation of alternate processes: Identify all the features in the given part and extract the design information such as geometry of the workpiece, dimensions, geometrical and dimensional tolerances etc. After recognition of features generate alternate processes to manufacture each feature [5].

2. Grouping (clustering) of machining features: Tool approach direction is a property of the process. Consider all the possible Tool Approach Directions (TADs) of the alternative processes of their respective features and group them according to same TADs as long as tolerance relation and features precedence are not violated.

3. Setup formation: After grouping the features, the setup is formed based on the tool approach direction. For example, a setup of +X tool approach direction.

4. Operation sequencing: Feature precedence network takes care of the sequencing of the operations in each setup.

5. Sequencing of the setups: The setups obtained are to be sequenced in such a way that tolerance and feature precedence relations are respected and get the optimal setup plan.

The benefits of computer aided setup planning are that it is able to give quick and optimal sequences of setups along with the information about machine and tool data to be used in the manufacturing process for the better quality of the product by attaining critical tolerance relationship between the features. Fixture planning and design is the next stage
of setup planning that helps to secure the workpiece during machining to meet the design specifications.

Although much focus has been given on fixture planning and design, there is still need to do more research. Fixtures are the work holding devices during machining. Fixture planning is required to have accurate and secured position of the workpiece at the time of machining. It is a decision making process and it consists of steps such as fixture configuration, conceptual design, detailed design and verification. After the setups are generated fixture planning is necessary for each setup. Therefore, in this thesis research fixture planning has been considered in integration with setup planning. The framework of fixture planning and design has been explained in the section 5.2.
2. LITERATURE REVIEW

In the last three decades there is high volume of research done on the CAPP system [1]. Setup planning is an important part in any CAPP system [6]. Many approaches have been considered in past research such as traditional, expert systems, hybrid graph etc. with different constraints such as cost, time, quality of the product, minimum setup, fixturing etc. The work done in the past on setup planning have considered many different approaches such as knowledge base, GA, ant colony optimization and constraints such as fixturing, cost, tolerances, tool change minimization, machine change cost etc. Therefore the literature review in this thesis has been classified in three categories: Multiple-constraints, fixture driven and tolerance analysis.

2.1 Multiple-Constraints

Setup planning by multiple constraint has been given much focus in optimizing the setup sequences, machining operations, cost associated with tools and machines. This section describe the papers that considered the mentioned objectives, their approaches and the limitations.

Kumar and Deb [7] proposed generation of optimal sequences of machining operations in setup planning using genetic algorithm (GA) technique which is based on minimizing the number of tool and setup changes. This paper constructs the GA as a new method for representing an operation as a distinct real number. Operation sequences are represented as chromosomes and they have tendency to loose feasible sequence due to randomness of GA. To minimize these losses, paper uses elitist model for selection of
chromosomes (operation sequences). This paper doesn’t consider the machining time and cost.

Li and McMahon [8] developed an algorithm to explore the search space more extensively by simulated annealing technique and optimized the process plan by considering machine cost, tool cost, machine change cost, tool change cost as objective functions. They have used three techniques i.e., processing, operation and scheduling flexibilities to support an algorithm or to generate process plans. Processing flexibility refers to possibility to perform an operation with an alternate machine and tool, operation sequencing flexibility refers to possibility of changing sequences of operations to be performed and scheduling flexibility refers to possibility of changing schedules to manufacture the parts and operations.

Guo et.al. proposes a method for optimization of operation sequencing in process planning by a particle swarm optimization approach. He concludes that particle swarm optimization can generate better results than genetic algorithm (GA) and simulated annealing (SA) techniques[9].

Liu et.al.[10] described the applications of ant colony optimization in process planning. He considers the minimizing cost for machining process as an objective function in process planning. He tries to solve this problem by converting manufacturing constraints, machine resources into constraint-based traveling salesman problem such as operations sets are mapped into city groups, processing costs of the cities into the weights of the cities etc. The cost of the product can lower down again if the scheduling of the
multiple parts production is optimized on the shop floor. However this paper doesn’t consider scheduling problems.

Neerukonda [11] has developed a framework for setup planning using object oriented approach. The objective of his thesis research is minimizing the number of setups. He considers only geometric data and tool orientation for clustering the operations in setups. His approach starts with sending operation data from CAM system to setup planning system, then new setups are created with operation sequencing in setup planning system and in the end data structure in the NX is updated based on result of setup planning. This thesis research doesn’t consider tolerance analysis.

Sormaz and Khoshenevis [12] consider process sequencing instead of feature sequencing because geometrical, technical and economical constraints are imposed by design and manufacturing. Process sequencing has been done in two stages. Process clustering based on tool orientation for each processes associated with the features is done in first stage. The processes which are associated with the features have same tool orientations are clustered together to. In the second stage, the best-first search algorithm is implemented to get the optimal process sequence.

Sormaz et. al.[13] proposed process plan model for manufacturing planning. The framework is integrated within the distributed system IMPlanner. This paper considers three dimensions: time/order, variability/alternatives and aggregation that are necessary in manufacturing planning. The developed model is useful to generate alternate process plans and select the suitable for the current status of operation.
2.2 Fixture Driven Setup Planning

In this approach fixture requirements are mainly considered as constraints. This approach doesn’t give only setup plan but also fixture configurations.

Stampfer[14] developed a model to generate the setup plan for box-shaped parts considering fixturing requirements. After sequencing the setups he tries to find out the suitable surfaces of box-shaped parts for locating and clamping during machining processes. It may happen sometimes that clamping elements obstruct the tool during machining. His system doesn’t give solution to avoid such a collision problem of clamping element and cutting tool during machining.

Attila et. al.[15] developed a system that gives setup plan and builds the fixture assembly for box shaped parts. The developed module tries to find such a setup in which all tolerance related features can be machined together. If the setup is not found they have to be machined in different setups. After the setup plans have been generated selection of fixturing elements from the database takes place. The research considers the constraint of tool collision with the fixture elements. The paper doesn’t consider the operation planning.

Joneja and Chang[16] developed a strategy to design minimum number of setups that can be fixtured using available machines and tools at shop floor. This research uses knowledge based approach. The developed process planner gives feasible fixturing configuration based on database of fixturing elements, fixture configuration planning (considered pre-defined locating and clamping points) and restraint analysis (considered
locating and clamping forces). However this paper doesn’t incorporate tolerance specified CAD models in the research.

Manafi and Nategh[17] used permutation-based approach for setup planning. They perform the sequencing of the setups first and next step is fixturing requirement for the setups. They developed an algorithm for setup sequencing based on two new sets of rules and these are technological and geometrical. Both the sets of rules are used for sequencing the machining features. Technological rules are concerned with locating and clamping surfaces, positioning of cutting tools, deformation of workpiece etc, while geometrical rules are concerned with geometrical interactions between the features. They state that during setup planning both sets of rules should be satisfied in order to sequence the setups. In case any conflicts may occur between these rules geometrical rules are ignored to resolve conflicts. However, in this research we developed rules to resolve cycle conflict between setups by considering both constraints geometric and technological.

Gologlu[18] has developed a setup planning module which simultaneously considers fixturing requirements. He considers tolerance relations between the features to generate feature precedences in setup planning while he decides fixture planning conceptually after the setup plan has generated. The system developed in this research handles only 2.5 D machining of prismatic parts on vertical milling machine.
2.3 Tolerance Analysis

Tolerance is one of the important criteria in setup planning since the accuracy of the manufactured part is always desired. Many setup planning approaches have taken tolerance analysis consideration into account.

Zhang and Lin[19] used a concept of “hybrid graph” which can be converted into directed graph by changing any two-way edge into one-way edge to show the tolerance relation between features, process relations and other information in CAD/CAPP/CAM. They used comprehensive principles such as feature grouping and setup sequencing for setup planning for 3-axis milling machines. They considered precedence among features in different setups for feature sequencing purpose but they did not consider precedence among features in a same setup. This paper doesn’t show the comprehensive analysis for both dimensional and geometrical tolerances.

Huang et.al.[20] also have used graph-theoretical approach for tolerance analysis in setup planning. Design specification of part and tolerance relation between features are represented as a graph. They have developed setup planning algorithm such that setup formation (based on TADs) will be done then datum selection and finally setup sequencing. Datum selection and setup sequencing was done based on a datum graph which shows the tolerance relations among features. However this paper doesn’t consider feature precedence for setup sequencing. They consider that it is always important to do tolerance analysis at an early stage and argues that tolerance charting (a tool for tolerance analysis) is always desirable after the setup plan has generated. This thesis research doesn’t consider datum selection but consider feature precedence for setup planning.
Kafashi[21] have used genetic algorithm approach for integrated setup planning and operation sequencing (ISOS). He considers feature precedence and tolerance relations in setup planning. He sequences the operation in order cost indices that means assigning best machine and tool for each operation so that to lower the cost. He mainly focuses on minimizing the cost of the product by considering different cost factors such as machine tool utilization cost, tool utilization cost, machine change cost, tool change cost, setup change cost and penalty cost.
3. PROBLEM STATEMENT

Although much research has been performed for the efficient setup planning there are still several loopholes. The major drawbacks in the current status of computer aided setup planning which aims to create optimal setups and fixture plans are as follows:

1. Lack of comprehensive tolerance analysis: Tolerance analysis is an important criteria in setup planning. It ensures the quality of manufactured products. Some of the researchers haven’t given much concern to tolerance analysis in setup planning and that could result in inflexible setup plans and stack-up errors. Moreover, some of the researchers have considered only one type of tolerances such as either geometrical or dimensioning. However, considering both will lead to an efficient setup plan and guarantee the better quality of the product.

2. Cycle generation within the setups: Most of the researchers haven’t considered this problem in their research. When the tolerance relations and feature precedence constraints are considered in the setup planning, there are most likely chances of creating the cycles between setups/features. For example, F1 should be machined before F2 according to tolerance criterion but feature precedence suggests to machine F2 before F1. Such situations should be handled properly in the setup planning.

3. Limited machine tools: Machine tools parameter is an important parameter in setup planning. Number of setups is directly related to the machine tools used for the machining prismatic parts. Most of the researchers have considered only NC machine, CNC machine, CNC 3-axis vertical milling machine for setup planning.
Other machine tools such as CNC 4-axis and 5-axis milling machines have received less attention, though they are important in manufacturing practise.

4. Lack of setup and fixture planning integration: Very few researches have made an attempt to integrate setup and fixture plan. Setup planning with the fixturing constraints was done only for the 3-axis milling machines. Most of the researchers have considered setup planning and fixture planning independently.
4. OBJECTIVE

In order to solve the problems discussed above, objective of this thesis is to use rule based system to develop setup and fixture plan which will accomplish following tasks:

1. Perform feature and process clustering (setup formation):
   Cluster or group the alternate processes for features according to shared tool approach directions (TAD) and geometrical and dimensional tolerances between the features to form setups.

2. Generate datum precedence (Tolerance Precedence Network - TPN):
   Consider the tolerance relations between datums and features and machine the datums first to minimize the stack-up errors, generate datum precedence in order to perform this task.

3. Generate operation sequencing within setups:
   Consider the feature precedence to sequence the operations within the setups.

4. Sequence of the setups:
   Sequence the setup for 3-axis and 4-axis vertical milling machine and get the optimal setup plan and avoid the cycles between the setups.

5. Perform fixture planning:
   Obtain the locating, clamping surfaces and points for the generated optimal setup plan.

6. Integrate rule based setup and fixture planning system with an IMPlanner (CAPP) system:
Necessary input to the rule based setup and fixture planning system consists of manufacturing features, processes and tolerances from an IMPlanner system. With this information, developed rules for the setup and fixture planning will fire to get the optimal setups and fixture plans.

The scope of this research is limited to only setup planning and determining the fixturing configuration.
5. METHODOLOGY

The research proposes to develop a framework of setup and fixture planning to get the minimum number of setups for a given prismatic part and finally obtain the fixture plan for the generated setups. The framework would be integrated with the existing CAPP system that is IMPlanner (Intelligent Manufacturing Planner) system. More details about the framework rule based setup and fixture planning (RB-SPS) module for integrating into IMPlanner system is given below.

5.1 IMPlanner System

IMPlanner system[13] is an existing CAPP research prototype system developed in the Industrial and Systems and Engineering department of Ohio University. IMPlanner system has been developed using object oriented approach. It provides the research bed for the students whose research interests are in design, manufacturing and process planning. It consists of important modules: a) process plan representation module (has knowledge about manufacturing processes and hierarchy), b) feature mapping module (it maps the machining features from CAD model), c) rule based process selection module (has manufacturing knowledge in terms of rules), d) process network module (has information about feature precedence), e) process visualization module (it helps visualizing and monitoring the selected processes in graphical user interface), f) setup planning module (current thesis research) and computer interfaces (integration of IMPlanner with CAD model and XML). An architecture of an IMPlanner system is shown in Figure 1 and highlighted at the bottom (Setup Planning) shows the current thesis research. The Setup Planning module has been integrated with IMPlanner system
since it requires part and processes information from IMPlanner system to be able to perform setup planning and that has been explained in section 5.2.

Figure 1 Architecture of IMPlanner system

The input to the IMPlanner system is feature based CAD (Siemens NX) model. The input CAD part is processed in the IMPlanner system to map/recognize the manufacturing features. IMPlanner system converts these mapped features into Jess facts and transfers them to the rule based process selection module. These facts run onto the process planning rules to get the alternate processes for each mapped feature which are used in RB-SPS system.
5.2 Setup and Fixture Planning Module

The architecture of setup planning module is shown in Figure 2. Setup planning module requires the input from IMPlanner system. IMPlanner provides necessary information such as mapped features, alternate processes, feature precedence, and tolerances on the features to the setup planning module to get the optimal setup plan. Setup planning module is being developed using rule based system ‘Jess’ (Java Expert System Shell) [22]. Jess is a part of Java which is a good tool to add knowledge in terms of rules and make reasoning about it.

Setup planning module consists of the tasks such as feature clustering, setup formation, operation sequencing and setup sequencing as shown. In Figure 2 the factors that influence decision in each of these tasks are shown under ‘constrains’ blocks in the figure.

The fixture planning module is also shown in Figure 2 and should be integrated with setup planning module to receive optimal setup plans as output. Once the setup plans are available finding fixturing configuration, conceptual, detailed design of fixture and verification tasks are performed in fixture planning. Some constraints such as shape of the workpiece and material influence fixture planning tasks. Integration of setup planning and fixture design tasks is particularly important in modern machines with advanced complex tooling and fixture options.

The first task of setup planning is Process Clustering: - Features and processes are clustered based on the shared tool directions. First, TADs for alternate processes of
features are generated. Then based on shared tool approach direction features with process are selected to cluster in order to prepare decision for setups.

The second task is Setup Formation: - Setups are formed of clustered features who have shared TAD and tolerance relations with each other. TAD is always the primary criterion for clustering the features in setups. Setup is a group of features with shared TAD. While clustering the features into setup based on TAD, geometrical and dimensional tolerance (GD&T) relations among features should also be considered for the better quality of the product. GD&T is defined by ANSI Y14.5M 1994 [23]. To attain a critical tolerance relationship between the features, such features should be clustered together if they share a common TAD. Similar to these considerations, authors [4] have proposed the following three methods for clustering the features:

Figure 2 Architecture of setup and fixture planning system
- Method 1: Group two features who have tightest tolerance relationship between each other. This method is always preferred since there are no stack-up errors.

- Method 2: Consider a feature as a datum and machine other. This method is less accurate and can be considered only when it is not possible to machine two features in same setup.

- Method 3: An intermediate datum is chosen and to machine two features in different setups. This method involves more stack-up errors.

We have used these three methods in the thesis with the following modifications and improvements and they are:

- Consider alternate processes for the features.

- Extended reasoning for tolerance related features as mentioned earlier.

Features with shared TAD should be clustered together in one setup. If the feature has multiple TADs, it is assigned to a single TAD depending upon its tolerance relation with other features. If the feature with the multiple TADs has tolerances but no relationship with any other feature then that feature is assigned to TAD that has maximum number of features. If the feature neither has tolerance nor datum then it will be assigned to the TAD that has maximum number of features.

We have considered a datum as a feature since the tolerance relationship is always specified between two features or between datum and a feature.

To form the 4-axis setups, we perform sub-tasks under setup formation:
a) Generate 4-axis setups: - First, we generate 3-axis setups for the given prismatic part. To make the 4-axis setups, we consider two 3-axis setups and generate 4-axis setups. The detailed methodology is explained in section 7.

b) Select 4-axis setups: - After the 4-axis setups are generated, we select 4-axis setups to minimize the total number of setups.

The third task is Operation Sequencing within the setup: - It is the step after setups of different TADs are formed. Operation sequencing is done based on the feature precedence and the datum precedence. In this research, the features whose machining doesn’t dependent on other features or the dependent features are already machined these features (their operations) are sequenced first and same procedure has been followed for other operations.

The fourth task is Setup Sequencing: - It is performed based on again feature precedence and datum precedence.

The third and fourth tasks are performed on setups for both 3-axis and 4-axis milling machines.

The detailed explanation and implementation of these tasks are given section 6.

Once the optimal setup plan is generated, the next step is fixture planning and design for the generated setup plan to secure the workpiece, ensures the stability and meets the design specifications.

The first step in the fixture planning and design is to identify Fixturing Configuration: - This step involves determining locating and clamping surfaces and points for each generated setup based on the geometry of the workpiece.
The second step is the Conceptual Fixture Design: - It involves determining fixturing layout and unit selection. Unit selection is basically choosing the type of fixture for the setups as well as determining the number of locating and clamping elements. The third step is Detailed Fixture Design: - Detailed fixture analysis is performed in this step. It involves conceptual design as well as determining material type, geometry for the fixtures. It may also involves analysis of force applied by locating and clamping points[24].

The scope of this thesis research is limited to only providing methodology for fixturing configuration. The rest of the tasks of fixture planning can be considered as future work.
6. SETUP PLANNING FOR 3-AXIS MILLING MACHINE

This section provides the methodology for setup planning for 3-axis (X, Y, Z) milling machine. In this thesis research 3-axis vertical milling machine has been used for setup planning. 3-axis horizontal milling machine isn’t considered in this research because it is not widely used in industries. Figure 3 shows the 3-axis vertical milling machine. Vertical 3-axis milling machines are widely used in industry since they are more flexible than other type 3-axis milling machines.

![Vertical 3-axis milling machine](image)

Figure 3 Vertical 3-axis milling machine [25]

The following setup planning tasks are performed for vertical 3-axis mill.

- Identification of tool approach directions for different manufacturing processes.
- Generate datum precedence (Tolerance Precedence Network - TPN) to machine the features.
• Setup formation based on TADs and tolerance relations among features to form the setups.
• Sequence the setups considering the FPN and TPN and also resolve the setup cycle conflicts.
• Sequence the operations within the setup considering FPN and TPN.

The detailed explanation, implementation and illustration for each task are given in the following sections.

6.1 Identification of Tool Approach Directions (TAD) for Feature Clustering

Tool approach direction (TAD) of the machining process axis is defined as the vector of the tool axis in the workpiece coordinates during machining of the features. TAD is considered as the property of process and process as the property of feature. TAD is a primary criterion for clustering the features in setups. For the prismatic parts there are total six tool approach directions. +X, -X, +Y, -Y, +Z and –Z on 3D mill [4]. Six tool approaches are common, but any other vector requires attention in fixture design. We have considered these tool approach directions as vectors in 3D space. Features are represented geometrically with their orientation vectors, and the first step is to convert those vectors into TAD objects for all alternate processes selected for each feature. For a good manufacturing practice group of features with processes having same TAD should be machined in a single setup. The tool approach directions for different features commonly found on prismatic parts depends on feature geometry and process kinematics. A manufacturing feature can have one or multiple processes [6] and for each process there can be single or multiple TADs.
Figure 4 shows the features with their geometry possible TADs and normal. TADs of each feature are explained below:

- The orientation of slab (Fig. 4a) feature is such that it has normal ‘n’ in Z-direction (on 3-axis mill). If the end-milling process is selected to machine the face/slab then its TAD will be ‘n’ vector. However, if the slab-milling process is selected then machining of slab can be performed in any direction which lies in X-Y plane.

- Blind-hole feature (Fig. 4b) has the bottom and can have only one TAD since hole can be accessible from only one direction. If the drilling process is selected then its TAD will be along hole-axis ‘ah’ vector.

- Through hole (Fig. 4c) has two TADs since it can be accessible from both directions ‘ah’ and ‘-ah’ vectors.

- Slot feature (Fig. 4d) has a normal and sweep vectors. Sweep vector is considered as orthogonal to the normal vector and along the slot volume. Fig. ‘4d’ shows the normal and sweep for blind-open-slot in thin lines. Slot also can be manufactured by few alternate processes. If the end-milling-slotting process is selected to machine the slot then its TAD will be ‘n’ vector (normal direction). If the side-milling process is selected then it has two TADs and those will be along the vectors which are the cross-product of normal and sweep vectors. Blind-open-slot (Fig. 4e) has total three TAD’s (one in normal direction and two in directions of cross-product of normal and sweep vectors). Closed-through-slot (Fig. 4g) has
total four TAD’s (one in normal direction and another opposite of normal. Remaining two in directions of cross-product of normal and sweep vectors)

- Pocket feature (Fig. 4f) has a normal direction and its profile. Blind-pocket (Fig. 4f) has only one TAD in normal direction and through-closed-pocket (Fig. 4h) has two TAD’s in normal and in opposite to normal direction.

In this research, we have converted this explanation into rules to assign feature and process to the TADs to create TAD objects and to cluster processes by TADs.

We have developed TAD identification rules for other features and they are as follows:
Blind Pocket
Through Pocket
Blind Open Slot
Through Slot
Slab
Blind Hole
Through Hole

TAD identification rules for blind-hole and through-hole features are explained in Figure 5 and Figure 6. For blind hole only one TAD is identified and it is through hole axis (Figure 4, b). For through hole feature two TAD’s are identified and they are positive and negative hole axis (Figure 4, c).

(defrule HoleTAD_Blind_Doesnt_EXIST
 (declare (salience ?*TAD_Prio*))
 (feature (name ?f1) (type hole) (bottom TRUE) (holeAxis $?hAxis))
 ?p <- (process (name ?pro) (type "drilling") (feature ?f1) (TAD $?tadp))
 (not (TAD (vector $?vTad &: (equal-list-list $?vTad $?hAxis))))
 =>
 (bind $?tad (concatenate-list $?hAxis)) (modify ?p (TAD (create$ $?tadp $?tad)))
 (assert (TAD (name $?tad) (features ?f1) (vector $?hAxis) (processes ?pro))))

Figure 5 TAD rule for blind-hole feature

Rule in the Figure 5 gives TAD for the blind hole feature process and it states that, if TAD of a vector $?hAxis of a feature $?fI doesn’t exist and the feature $?fI is a hole who has a bottom and it has process $?pro which is ‘drilling’ then assign a new TAD $?tad of the feature $?fI of vector $?hAxis since drilling must be done along hole-axis.
Rule in Figure 6 identifies the TAD for the through hole. Through hole has two TADs ‘ah’ and ‘-ah.’ Positive TAD will be identified according to similar rule (developed for through hole feature) as mentioned in Figure 5 and negative TAD will be identified according to the rule showed in Figure 6 which is basically same as that of positive TAD except negative co-ordinates of the vector which are mentioned in the rule.

```
(defrule HoleTAD.Through.NegativeNormal.Doesnt.Exist
  (declare (salience ?"TAD_Prio")
  (feature (name ?f1) (type hole) (bottom FALSE) (holeAxis $?hAxis))
  (?p < (process (name ?pro) (type "drilling") (feature ?f1) (TAD $?tadp))
  (not (TAD (vector $?vTad&: (equal-list-neg-list $?vTad $?hAxis))))))
=>
  (bind ?tad2 (concatenate-neg-list $?hAxis))
  (modify ?p (TAD (create$ $?tadp ?tad2)))
  (assert (TAD (name ?tad2) (features ?f1) (vector (negative (nth$ 1 $?hAxis))
    (negative (nth$ 2 $?hAxis)) (negative (nth$ 3 $?hAxis))) (processes ?pro))))
```

Figure 6 TAD rule for through-hole feature

Additional rules have been developed when the same TAD already exists which is shown in Figure 7. This rule states that if there is a feature ?f1 with a process ?pro which has a TAD ?tad which is not a member of $?tadp (list of TADs in process templates) then add the ?tad into the list of $?tadp into process template. At the same time our rule also checks for the processes and features if they are the member of $?processes (list of processes in TAD template) and $?features (list of features in TAD template) respectively, if not then remember that feature into $?features in TAD and process into $?processes into TAD template.

Figure 8 shows the sample facts that will run on the TAD rule of blind-hole and through hole features. There are two holes Hole1 (blind-hole with bottom) and Hole2
(through-hole with no bottom). Hole1 has drilling1 process and (0, 0, 1) hole axis vector while Hole2 has drilling2 process and (0, 1, 0) hole axis vector. The result is shown in Figure 9. It shows that, facts f-0 to facts f-3 are the input facts. Facts f-0 and f-1 are the given feature and process instances respectively for feature HOLE1. Facts f-2 and f-3 are the feature and process instances for feature HOLE2. Facts f-4 to f-6 are the output facts which show generated TADs for both features. Fact f-4 indicates that TAD (0,-1,0) has been identified for feature HOLE2. Since the feature HOLE2 is through hole alternate TAD has been identified by fact f-5 which is (0, 1, 0). Since the feature HOLE1 is blind hole only one TAD is identified that is (0, 0, 1). All these TADs have been identified according to hole axis orientations.

```lisp
(defrule HoleTAD.Blind.Does.Exist ; Blind Hole when a TAD exists
  (declare (salience "TAD_Prio")
  (feature (name ?f1) (type hole) (bottom TRUE) (holeAxis $?hAxis))
  ?p <- (process (name ?pro) (type "drilling") (feature ?f1) (TAD $?tadp))
  ?t <- (TAD (name $?tad&: (not (member $?tad $?tadp))) (features $?features) (vector $?vTad&: (equal-list-list $?vTad $?hAxis)) (processes $?processes))
  (test (not (member $?pro $?processes))); to avoid infinite loop
  (test (not (member $?f1 $?features))))
=>
  (modify ?p (TAD (create $?tadp $?tad))) (modify ?t (features (create $?features $?f1)))
  (modify ?t (processes (create $?processes $?pro))))
```

Figure 7 TAD when it exists

For ex. if a feature HOLE3 is a blind hole with hole axis (0, 0, 1) and with process drilling3 then its TAD should be (0, 0, 1) but if such a TAD exists then assign the feature HOLE3 to the existing TAD and remember the process in the existing TAD template.
Once all the TADs are identified for all processes of the features of the part, next task of the setup planning is of grouping processes based on tolerance relation to make setups.

6.2 Feature and Process Clustering by TAD and GD&T

In the current work, method 1 (section 5.2) has been based to cluster the processes of features into setups with significant extensions and improvements. The book [4]
doesn’t consider the alternate processes for the features and considers TAD as a property of a feature. In our research we consider the alternate processes for the features and TAD as a property of process. In order to explain the reasoning we use a hypothetical example shown in Figure 10 and Table 1, which show the tolerance relationship between the features and TADs.

Feature F1 has no tolerance. F1 feature is a primary datum for F2, F3 and F6 features. F2 and F3 have tolerance type perpendicularity and tolerance value 0.1 and 0.01 respectively with respect to feature F1. F6 has parallelism tolerance with 0.01 tolerance value. F4 has a tolerance type parallelism and has tolerance value 0.1 with respect to feature F2. F5 has tolerance type circularity and has tolerance value 0.5 with respect to feature F4. F6 is a secondary datum for F3 has a parallelism tolerance with 0.01 tolerance value. F6 is a tertiary datum for F3 has an angularity tolerance with 0.01 tolerance value.

Figure 10 Tolerance relationship between features for example
We have assumed some processes with TADs for above features to test our feature clustering rules based on tolerance relation and same TAD.

Table 1 shows that each feature has one or more processes and each process has one of more TADs. Features with tolerance relationships are also shown.

We have developed following feature and process clustering rules based on TAD and tolerance relations:

- Rule 1: Clustering of the features which have tolerance relations.
- Rule 2: Adding a feature that has tolerance relation with other feature which is already clustered.
- Rule 3: Assigning the feature that has tolerance and NO datum to a setup.
- Rule 4: Assigning the feature which has neither tolerance nor datum to a setup.

Our developed rules for clustering the features give first priority to those features that have tightest tolerance relations. Expected reasoning in the example is as follows.

First, F3, F7 and F6 should be grouped with the feature F1 since they have shared ‘TAD 3’ and have tightest tolerance relationship (tolerance value 0.01) with feature F1. This clustering will be done by rule 1.

Second, the next step is to cluster those features that have loose tolerance relations. In this example, F2, F4 and F5 will try to group together and form a setup because they share a common ‘TAD 2’. This clustering will be done by rule 1.

Third, feature F8 has tolerance but no datum. Even though it has tightest tolerance it will be clustered in the setup which consists of maximum features. Since feature has ‘TAD 2’, it will look up for the setup which has maximum number of features of ‘TAD
Therefore, feature F8 will cluster in the setup which has feature F2, F4 and F5 and same TAD as that of feature F8. This clustering will be done by rule 3.

Table 1 Processes and TADs for features in example

<table>
<thead>
<tr>
<th>Features</th>
<th>Tolerance with Datum</th>
<th>Processes</th>
<th>TADs</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td></td>
<td>P1</td>
<td>TAD1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P2</td>
<td>TAD1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P3</td>
<td>TAD1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P4</td>
<td>TAD4</td>
</tr>
<tr>
<td>F2</td>
<td>Perp. 0.1 with F1</td>
<td>P5</td>
<td>TAD2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P6</td>
<td>TAD3</td>
</tr>
<tr>
<td>F3</td>
<td>Perp. 0.01 with F1</td>
<td>P6</td>
<td>TAD3</td>
</tr>
<tr>
<td></td>
<td>Para. 0.01 with F6</td>
<td>P7</td>
<td>TAD4</td>
</tr>
<tr>
<td></td>
<td>Ang. 0.01 with F7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F4</td>
<td>Para. 0.1 with F2</td>
<td>P8</td>
<td>TAD2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TAD5</td>
</tr>
<tr>
<td>F5</td>
<td>Circu. 0.5 with F4</td>
<td>P9</td>
<td>TAD2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TAD4</td>
</tr>
<tr>
<td>F6</td>
<td>Para. 0.01 with F1</td>
<td>P10</td>
<td>TAD3</td>
</tr>
<tr>
<td>F7</td>
<td>Ang. 0.01 with F1</td>
<td>P11</td>
<td>TAD3</td>
</tr>
<tr>
<td>F8</td>
<td>Str 0.01 with none</td>
<td>P12</td>
<td>TAD2</td>
</tr>
<tr>
<td>F9</td>
<td>None</td>
<td>P13</td>
<td>TAD2</td>
</tr>
</tbody>
</table>

Finally, Feature F9 (has no tolerance and datum) will be assigned to a setup which has maximum features and same TAD as that of F9. Therefore, F9 will be assigned to setup of ‘TAD 2’ since its TAD is ‘TAD 2’. This clustering will be done by rule 4.

We have implemented all those rules in Jess and rule 1 is shown in Figure 11.
Figure 11 Rule to cluster features that have tolerance relationship between them

The rule in Figure 11 states that, if a feature $f_1$ has tolerance relation with primary datum $pd$. Feature $f_1$ has one or multiple tolerances and one of them is $tol$ of tolerance value $tv$, process $pro$ and that process $pro$ has one or multiple TADs and one of TADs is $tad$. Primary datum $pd$ which is also another feature having the process $pdpro$ and $pdpro$ has multiple TADs and one of them is same as that of feature $f_1$ that is $tad$. We have inserted a condition such that if none of the other feature whose tolerance value is less than $tv$ then feature $f_1$ and $pd$ should be grouped together in a setup of TAD $tad$. On the RHS side of the rule states that $pd$ and $f_1$ features are grouped together in ‘decideFeatures’ slot of TAD template. Moreover, respective processes of both features should also be grouped together and it has been done in ‘decideProcesses’ slot of TAD template. Other rules will be described in section 9.
Figure 12 shows the clustered features based on shared TAD and tolerance relations. Two clusters have been formed that means two setups are formed in which all the features can be machined. The corresponding result in the Jess is shown in Figure 13.

Since the feature F1, F2, F3, F6 and F7 have tightest tolerance relation and they share a common TAD, hence they have clustered together in a setup ‘TAD3’ while F2, F4 and F5 have loose tolerance relation (tolerance value 0.1) they form a cluster ‘TAD2’ since they share a common TAD. F8 and F9 are the independent features without any tolerance relations therefore these features look for the setup of the same TAD and maximum features. ‘TAD2’ is the setup of the same TAD and maximum features hence F8 and F9 get clustered into the ‘TAD2’ setup. From the Figure 13 it can be seen that out
of **Five** possible setups, the rule based system selected only **Two** setups in which all the features are manufactured.

<table>
<thead>
<tr>
<th>Rule</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>f-24</td>
<td>(MAIN::TAD (name TAD1) (features F1) (processes P1 P2 P3) (decideProcesses ) (decideFeatures ))</td>
</tr>
<tr>
<td>f-25</td>
<td>(MAIN::TAD (name TAD2) (features F1 F2 F4 F5 F8 F9) (processes P1 P2 P5 P8 P9 P12 P13) (decideProcesses P12 P5 P8 P9 P13) (decideFeatures F8 F2 F4 F5 F9))</td>
</tr>
<tr>
<td>f-26</td>
<td>(MAIN::TAD (name TAD3) (features F1 F2 F3 F6 F7) (processes P3 P5 P6 P10 P11) (decideProcesses P10 P6 P3 P11) (decideFeatures F6 F3 F1 F7))</td>
</tr>
<tr>
<td>f-27</td>
<td>(MAIN::TAD (name TAD4) (features F1 F3) (processes P4 P7) (decideProcesses ) (decideFeatures ))</td>
</tr>
<tr>
<td>f-28</td>
<td>(MAIN::TAD (name TAD5) (features F4) (processes P8) (decideProcesses ) (decideFeatures ))</td>
</tr>
</tbody>
</table>

**Figure 13** Setup formation based on tolerance relation and TAD

### 6.3 Datum Precedence (Tolerance Precedence Network - TPN)

Datum precedence is another constraint in the setup planning. Datum is an imaginary or real plane used to locate the part [19]. Datum can also be considered as a feature. Geometrical and dimensional tolerances are always specified between feature and a datum. The imaginary plane on which the component lies during machining is called primary datum. Secondary datum is perpendicular to the primary datum and tertiary datum is perpendicular to both of them [4]. Machining all the datums first is a good manufacturing practice and doesn’t cause stack up error. Datum precedence shows which datum should be machined prior to which datum. The current research shows the datum precedence in the form of graph which is generated using Jess and IMPlanner tool. Consider the same example mentioned in Figure 10. The rule which generates a graph is shown in Figure 14.
The rule states that when feature \( f_1 \) has tolerance \( t \) and primary datum \( d \) then add datum \( d \) in the ‘previous’ slot of template feature whose name is \( f_1 \) only when the \( d \) is not in the list of \( l_d \). At the same time, feature \( f_1 \) is added to the ‘next’ slot of template feature whose name is \( d \) only when \( f_1 \) is not in the list of \( l_f \). This indicates, machine the datum \( d \) first and then feature \( f_1 \). We have developed similar rules for secondary and tertiary datum.

```
(defrule make-primaryDatum-precedence
  (?f <- (feature (name ?f1) (tolerance ?t) (previous $?ld)))
  (tolerance (name ?t) (type ?type) (primaryDatum ?d))
  (?df <- (feature (name ?d &. (not (member$ ?d $?ld)) (next $?lf)))
  (test (not (member$ ?f1 $?lf)))
  =>
  (modify ?df (next (create$ $?lf ?f1)))
  (modify ?f (previous (create$ $?ld ?d))))
```

Figure 14 Rule for datum precedence

Figure 15 Datum precedence (TPN)
In the graph (Figure 15), the feature that has an arrow pointed towards it, which indicates it should be machined after the feature from where an arrow is originating. F1 is a datum for feature F2, F3, F6 and F7 therefore F1 should be machined first. F6 and F7 is a datum for F3 therefore F6 and F7 should be machined before F3. Other features are machined in the same fashion.

In order to prepare results of this step for the steps we will now show application of the described procedure on a real example part. This example shows the generation of setups for Slider part based on only shared tool approach direction and tolerance relations. Also shows the generation of tolerance precedence network for Slider part. The Slider part will be completely shown later in section 10. For now we only show features in Figure 16.
At this point the input to the setup planning framework is actually given in the form of manually created .CLP files (Jess files) which comprises of all the design information such as features, processes and tolerances for a given part. The first step upon loading the part data is generation of tolerance precedence network which is shown in Figure 17.

![Figure 17 Datum precedence (TPN) for Slider](image)

Tolerance precedence network shows (Figure 17) that feature D should be machined before Hole 15, Hole 16 and Hole 17. However since there is a connection between A and D with an arrow pointing towards feature A, that means feature A should be machined before D.

Figure 18 shows the portion of the result of Slider part. Features Hole 15, Hole 16 and Hole 17 have tolerance relationship with feature A. All of them have single process
and each process has one or more TADs. Since all Hole features have tolerance relationship with feature A, all of them should be clustered together to form one setup if they share common TAD. All Hole features share common TAD (0 -1 0) shown in broken elliptical shape but feature A doesn’t have TAD (0 -1 0). However, TAD (0 1 0) in solid elliptical shape is a common TAD for all of them. Hence single setup ‘Setup 1’ is formed for all four features of TAD (0 1 0). Similarly other features form setups accordingly.

Figure 18 Setup formation of Slider part

Upon execution of all rules mentioned earlier Figure 19 shows the final result of feature clustering of Slider part. Fact-60 to fact-65 are the generated setups of their respective TADs. Features, vectors and processes are also shown associated with these TADs.
Fact f-79 to f-83 are the setups formed for the 3-axis vertical milling machine for six different TADs. Each setup has its own vector and decided features and processes according shared TAD and tolerance relations. Thus only five setups are formed for Slider part.

Setup sequencing based on Feature Precedence Network (FPN) and Datum/Tolerance Precedence Network (TPN) is explained in the next section.

6.4 Setup Sequencing

After the feature clustering and setup formation based on shared TAD and tolerance analysis, setups are sequenced. FPN and TPN are the constraints for the setup sequencing. Feature precedence states that which feature should be machined prior to other feature. Tolerance precedence shows the similar case that is which feature (datum) should be machined prior to other feature based on tolerance relations.

Feature precedence network can be generated from IMPlanner for mechanical parts designed in Siemens NX [26] (it will be shown later on case studies). The generated FPN considers the geometric and technological constraints between the features. Our
setup planning module considers an augmented FPN which is a combination of FPN and TPN. Our setup planning module is capable to generate an augmented FPN with the help of IMPlanner and setup planning integration. Since the FPN and TPN are combined there are most likely chances of cycles occurring precedence which cause conflicts between setups [2]. Cycle between setups can be shown with an example in which we have those precedences $S_1 \rightarrow S_2$ and $S_2 \rightarrow S_1$. Based on the geometrical and technological constraint FPN may say that setup 1 should be machined prior to setup 2 but based on the TPN setup 2 should be machined before setup 1. To avoid such a problem, our approach is to break down setups such that minimum tolerance violation can be seen and minimum number of setups can be obtained. Prior to breaking down setups, minimum tolerance violation may exists because clustering is purely based on shared TAD’s and tolerance relations. If two features with tolerance relations do not have shared TAD, they both are assigned to different setups and in this case minimum tolerance violation may occur.

We have considered two cases of setup conflicts in this thesis. Three feature conflicts and Four feature conflicts.

### 6.4.1 Three Feature Conflict

Case 1 has been shown in Figure 20. Features F1 and F2 are the part of setup 1. F1 is the previous feature for F3 and F2 is the next feature for F3 which is in a setup 2. F2 has smaller number of ‘next’ features while F1 has larger number of ‘next’ features because F1 may be a datum feature for other features. In short, based on F1-F3 relation, setup 1 should be machined prior to setup 2 but based on F2-F3 relation setup 2 should be machined prior to setup 1, hence this is the conflict situation between two setups. To
avoid the conflict between these two setups, F2 (has less number of next features) feature should be removed from setup 1 and assigned to newly created setup that is setup 3. In this way, cycle precedence or conflict has successfully been eliminated and new sequence of the setups have achieved. Sequence of the setups for machining will be Setup 1 → Setup 2 → Setup 3. Once the setup sequencing is performed without any conflicts, the next step is operation sequencing within the setups according to FPN and datum precedence.

![Figure 20 Three features conflicts](image)

### 6.4.2 Four Features Conflict

In the case of four features setup conflicts (Figure 21) we have applied the same reasoning as above. In this case features F1, F2 are the part of setup 1 and F3, F4 are the part of setup 2. F3 is the next feature of F1 and F2 is the next feature of F4. To avoid the setup conflict in this case, one of the four features that smaller less number of next features should be removed from its own setup and should be assigned to newly created setup that is setup 3. Feature F2 has less next features therefore it was removed and
assigned to a different setup that is setup 3. In this case, the sequencing of setup will be 
Setup 1 ---→ Setup 2 ---→ Setup 3.

There are other possible conflicts situation such as combined three-four feature 
conflict or there are three or feature conflicts with multiple setups. Since, three and four 
feature conflicts are frequently occurred therefore this thesis research demonstrates only 
two possible cases.

If there is case such that four or more setups have cycle conflicts, for example, 
S1→S2→S3→S4→S1. In this case setup cycle conflicts will be resolved with one of two 
reasoning (three or four feature conflict resolution).

We have developed a rule for the setup sequencing and elimination of conflicts 
between the setups. The rule in Figure 22 considers the case 1 that is three features 
conflict.
The rule states that there are two features F1 and F2 in first setup have precedence with the feature F3 in second setup such that F1 is the previous feature for F3 and F2 is the next feature for F3. In that case our rule removes that feature from first setup that has less number ‘next’ features and creates a new setup for that feature to avoid the cycle precedence. The reason of excluding the feature that has less ‘next’ feature is that, tolerance related features/datum may have maximum ‘next’ features and excluding them from original setup and assigning them to a new setup may cause more tolerance violation or it may cause more stack up error. Excluding features that have less ‘next’ features may cause fewer stack up errors. In the same fashion we developed a rule to resolve four feature conflicts. Resolving three and four feature conflicts in the setup planning model is completed by executing those rules. Illustration of given precedence is shown on Slider example.
For setup sequencing, we generate an augmented FPN from an integration of IMPlanner and setup planning module which is a combination of FPN and TPN to have more constraints to identify cycle precedence and produce sequence of the setups. Figure 23 shows an augmented FPN with the setups generated by the setup planning module.

![Setup conflicts in augmented FPN](image)

Setups are shown in color coded circles and each color code indicates different setup. From the Figure 23, it can be seen that setup planning module has created 5 different setups according to shared TAD and tolerance relations where all the features can be machined. But there are some conflicts within the setups that have occurred due to augmented FPN and setup formation based on shared TAD. These conflicts have been shown by arrows. There are cyclical precedence between Setup 1 and Setup 2. Therefore
our setup sequencing rule (Figure 22) works to resolve these conflicts and break the setups in such a way that the minimum violation of tolerances are occurred. The final result of setup sequencing (without conflicts) is shown in Figure 24. The augmented FPN was generated by setup planning module. However, the features are re-arranged manually to show the setups for 3-axis vertical mill. Different colors show different setups (setups of different vectors).

![Figure 24 Setup conflict resolution](image)

The setup sequencing rule, searches for the 2 features from one setup and one feature from another setup such that there is precedence and removes the feature from first setup that has less number of next features to form new setup.
Name of some features (in Figure 16) are changed here according to the existing CAD model of Slider. The reason behind this earlier setup planning was performed manually (some of the feature’s names were assumed) without integration with CAD since the focus was on procedure and after integration of setup planning with CAD the names of all features were imported from CAD. The new names of features are:

Feature A :- RECTANGULAR_POCKET(2)
Feature F2 :- RECTANGULAR_POCKET(3)
Feature F3 :- RECTANGULAR_SLOT(8)
Feature F1 :- RECTANGULAR_POCKET(7)
Feature C :- RECTANGULAR_POCKET(4)
Feature D :- RECTANGULAR_POCKET(6)

From the Figure 23, it can be seen that Hole 29 (Setup 1) is the ‘next’ features of Slot 9 (Setup 2) and Slot 8 (Setup 2) feature is ‘previous’ for feature Pocket 2 (Setup 1) (this is a case of case 2 shown in Figure 21). In this case our rule searches for the feature among all features Hole 29, Slot 9, Slot 8 and Pocket 2 that has less number ‘next’ features. Since the Hole 29 has less ‘next’ features than the other three features, therefore another setup is created ‘Setup 1-1’ where Hole 29 becomes the candidate of it (Figure 24). Similarly, features Hole 27, Hole 26 and Hole 28 are clustered into different setup 1-1. The result of setup sequencing obtained from setup planning module is shown in Figure 25.
Facts f-74 to f-78 are the total five setups generated by setup formation step before the setups breaking or before the resolution of cycle conflicts. When the setup-sequencing rule was fired to eliminate the cycle precedence facts f-79 (Setup-TAD:0:1.0:0.0:1) is the new setup created with the respective features shown above. Setup (Setup-TAD:0:1.0:0.0:1) is extracted from the setups (Setup-TAD:0:1.0:0.0). The reason for which the setups were broken is given by the ‘cause’ slot.

Finally, minimum setups are formed for ‘Slider’ mechanical part and ensure machining of tight tolerance related features in same setup as well as following FPN and TPN constraints.
6.5 Operation Sequencing within the Setups

After the final setups are created operation sequencing is done within each setup. Features/operations are ordered or sequenced based on ‘next’ and ‘previous’ features relationship (augmented FPN) in the setup planning module. Firstly, our operation sequencing rules sequence such feature/s within the setup that has no ‘previous’ feature/s. If there are multiple features within a setup, our rules sequence those features first who are datum of other features and then rest of the features (independent features with no tolerance relation and FPN constraints). Following Figure 26 shows the operation sequencing rule for those features who have no ‘previous’ features.

```lisp
(defrule operation-sequencing-within-setups
  (declare (salience "operation-sequencing")
    ?fm <-(featuresMachined (feature $?featuresMachined))
    ?s <-(setup (name ?name) (features $?features) (eligible $?eligible))
    (feature (name $?f1) & (and (member$ $?features) (not (and (member$ $?f1 $?featuresMachined)) (member$ $?f1 $?eligible))))
    (prev $?previous1) (next $?next1) (test (subsetp $?previous1 $?featuresMachined))
  )=>
    (modify ?fm (feature (create$ $?featuresMachined $?f1)))))
    (modify ?s (eligible (create$ $?eligible $?f1))))
```

Figure 26 Operation sequencing within setup when no previous features

The rule states that when there is a setup of 3-axis machines of name ?name with the list of features $?features and feature $?f1 which is a member of the list $?features and not yet sequenced (i.e., not a member of $?featuresMachined) within the setup. A condition (test) when list of previous features $?previous1 is a subset of $?featuresMachined (no previous features for feature $?f1) then add that feature into sequenced features list $?featuresMachined and make that feature eligible for that
particular setup. In such a way features from particular setup with no previous features are made eligible and place next to previously sequenced feature. Thus it makes order for the features to be machined in ‘eligible’ slot of that particular setup. Secondly, we have developed another rule which sequences or order those feature those are available after the features those are sequenced. For example, F1 \(\rightarrow\) F2 \(\rightarrow\) F3 is the FPN within a setup. The second rule of operation sequencing will sequence F2 because it is available for machining since F1 was already sequenced before (F1 doesn’t have ‘previous features’) by a rule showed in Figure 26. The result of operation sequencing within setup for ‘Slider’ part is shown in Figure 27. Facts f-74 to facts f-79 are the finalized setups for ‘Slider’. Each setup has name, vector, features and processes associated with them. Slot ‘eligible’ shows the sequenced features (selected in Bold). For example, for the setup “Setup-TAD:0:0:1" features are ordered as HOLE31 \(\rightarrow\) HOLE24 \(\rightarrow\) POCKET14 that means from POCKET14 and HOLE31 any one of them can be machined first since both of them have no previous features (Figure 25) but our rule based system ordered HOLE31. When the HOLE31 was ordered POCKET14 and HOLE24 get eligible for machining or sequenced because HOLE24’s previous feature HOLE31 was already sequenced and POCKET14 has no ‘previous’ features. In that case, our rule based system has sequenced POCKET14 and finally HOLE24. In this way features from each setup got sequenced in ‘eligible’ property of setups.
Figure 27 Operation sequencing within setup for Slider
6.6 Setups Precedence and Ordering

Setups ordering are done after final setups are obtained for the mechanical part. In this thesis research, setups ordering is done with the help of setup precedence or ‘next’ and ‘previous’ relation of setups. Setup precedence shows which setup should be machined prior to which setup. We have developed a rule to generate setup precedence and followed by setup ordering.

6.6.1 Setup Precedence

Finalized setups should be machined in such a way that they follow TPN and FPN. Setup precedence in this research was developed considering an augmented FPN as constraints. We developed the following rule which generates setup precedence.

```
(defrule next-setups
  (declare (salience ?"next-previous-setups")
    ?s <- (setup (name ?name1) (features $? ?f1 $?) (next $?next))
    (feature (name ?f1) (next $? ?n1 $?))
    (setup (name ?name2 & (neq ?name2 ?name1)) (features $? ?n1 $?))
    (test (not (member$ ?name2 $?next))))
  =>
    (modify ?s (next (create$ $?next ?name2)))
)

(defrule previous-setups
  (declare (salience ?"next-previous-setups")
    ?s <- (setup (name ?name1) (features $? ?f1 $?) (previous $?previous))
    (feature (name ?f1) (previous $? ?n1 $?))
    (setup (name ?name2 & (neq ?name2 ?name1)) (features $? ?n1 $?))
    (test (not (member$ ?name2 $?previous))))
  =>
    (modify ?s (previous (create$ $?previous ?name2)))
)
```

Figure 28 Rule for setup precedence
The basic reasoning we applied to generate setup precedence is that; between two setups, setup 1 and setup 2, if a feature from setup 1 has a ‘next’ feature in setup 2 then setup 1 should be machined prior to setup 2.

Based on this ‘next’ and ‘previous’ relation setup precedence is defined. The ‘next-setups’ rule indicates that when there are two different 3-axis mill setups $\text{name1}$ and $\text{name2}$, each of them have list of features (for machining). Feature $f1$ is a member of list of features of setup $\text{name1}$ that has ‘next’ feature $n1$ which is a member of list of features of another 3-axis setup $\text{name2}$, Then add setup $\text{name2}$ in the ‘next’ list of setup $\text{name1}$ if it is not already added. Similarly, another rule ‘previous-setups’ works in same way where feature $f1$ from setup $\text{name1}$ has ‘previous’ feature $n1$ in the list of features of setup $\text{name2}$ then add setup $\text{name2}$ in the ‘previous’ list of setup $\text{name1}$ if it not already added.

The Figure 29 shows the setup precedence graph tool. The arrow is pointing towards setup ‘Setup-TAD: -1:0:0’ that means ‘Setup-TAD: -1:0:0’ should be machined after setup ‘Setup-TAD:1:0:0’. After the setup precedence is generated, setup ordering or ranking is done based on setup precedence.
6.6.2 Setup Ordering

Setup ordering is performed to order/rank generated setups for the machining purpose. Setup ordering is done in the same fashion as operation sequencing is performed in 0. However, in case of setup ordering (when more than one setup is available for machining), the setup will be ordered based on the number of ‘next’ features for one of the features associated with that particular setup. For general manufacturing practice, the setup with a feature which has maximum number of ‘next’ features than the other setup, should be machined $1^{st}$ because machining a setup with maximum number of ‘next’ features is always desired since the datum feature should always be machined first. To apply these reasoning we have developed 2 rules; to order setups (with no previous list of setups) and to order setups with previous list. Following rule shows the ordering of setups which have no ‘previous’ list of setups.
The rule from Figure 30 is developed to order setup when there is no previous setup associated with it.

```
(defrule order-setups-with-no-previous-list
  (declare (salience ?"order-setups")
    (?s <- (setupsMachined (setup $?setups)))
    (?s1 <- (setup (name ?name1&. (not (member$ ?name1 $?setups))) (features $? ?f1 $?) (previous $?previous&: (eq $?previous (create$)) (order nil))
      (feature (name ?f1) (next $?next1))
      (setup (name ?name2&: (and (neq ?name2 ?name1) (not (member$ ?name2 $?setups)))))) (features $? ?f2 $?))
    (not (feature (name ?f2) (next $?next2&: (> (length$ $?next2) (length$ $?next1))))))
  (=>
    (bind ?"order" (+ 1 ?"order")
      (modify ?s1 (order ?"order")
        (modify ?s (setup (create$ $?setups ?name1))))))
```

Figure 30 Rule to order setups for 3-axis mill

It states that, if a 3-axis setup ?name1 which is not ordered or machined (not member of $?setup list) has a list of features with previous list nil. A feature ?f1 which is a member of list of features associate with setup ?name1 and has list of next features $?next1. When a feature ?f2 from other setup ?name2 has less next features than feature ?f1 then order setup ?name1 as ‘1’ and remember the this setup in setupsMachined, in that case the rule will consider other setups except ?name1 for ordering. We have developed similar rule with same reasoning which considers the setups with ‘previous’ list.

Figure 31 shows ordered setups for Slider obtained from rule based system.
<table>
<thead>
<tr>
<th>Figure 31 Ordered setups for Slider</th>
</tr>
</thead>
</table>

"Setup-TAD:1:0:0" is given order 1 since this setup has no 'previous’ setup list.

"Setup-TAD:0:1:0" is given order 2 since it has no 'previous’ list (refer Figure 29). Similarly other features are given orders accordingly; "Setup-TAD:0:0:1" – order 3, "Setup-TAD:-1:0:0" – order 4, "Setup-TAD:0:-1:0" – order 5, "Setup-TAD:0:1:0-1" – order 6.
Finally, setup planning for 3-axis milling machine is completed by performing above described steps for formation of setups – Identification of TADs, feature and processes clustering by shared TADs, and tolerance relation, setup sequencing and ordering based on augmented FPN and finally operation sequencing within the setups based on augmented FPN. Figure 31 shows the final setups for 3-axis milling machine.
7. SETUP PLANNING FOR 4-AXIS MILLING MACHINE

This chapter provides methodology for setup planning for 4-axis milling machines. The most common type four-axis milling machines are machines with rotary table as shown in Figure 32. It has three linear axes x, y, z and rotary axis b.

![Horizontal 4-axis milling machine](image)

Figure 32 Horizontal 4-axis milling machine [27]

In case of horizontal 4-axis milling machine the cutting tool can machine the 4 faces of part (prismatic) in one setup when part is fixture on rotating table ‘b’ (cutting tool direction z-axis). Few other types of horizontal 4-axis milling machines are shown in Figure 33. For those types three linear axes may move tool head or workpart. They also have one rotational axis A (around x) or B (around y), with a usually rotating tool head and B rotating the machine table.
The Figure 34 shows the 4-axis vertical milling machine. Vertical 4-axis milling machines are generally used to machine plate shaped parts. In the Figure 34 4-axis milling machine has vertical spindle rotating along Z axis. X, Y, Z are the three linear CNC axes and A’ is a rotary table along X-axis. Four axis milling machine extend processing capability for milling operations in two ways:

1) Operation that do not require tool rotation during milling.
2) Operations that require tool rotation during milling (for example, milling of helix groove on a shaft).

For the first type of operations 4-axis mill provides benefit of minimizing workpiece setup since tool can approach it from different orientations. For ex. Using rotation axis b in Figure 32.

For those operations TAD of a process does not change and setup needs to accommodate all operations for which their TAD is orthogonal to the rotational axis b. For the second type of operations, four axis mill is required in order to machine desired designed part geometry.

For these operations TAD of the process becomes dynamic since it changes respective to the workpiece coordinates during machining. Those operations are characterized with not a single TAD vector, but by a set of required TAD vectors for tool/part kinematics and thus needs to be calculated before setup planning.

Based on this consideration we have developed a methodology to generate 4-axis setups for the first type of milling operations which is applicable for various
configurations of 4-axis milling machines. The methodology is described in the following sections.

Figure 33 Other types of 4-axis milling machines [28]
7.1 Generation of 4-Axis Setups

From the above discussion, we have mentioned that on horizontal 4-axis milling machine four faces can be machine at one in block shaped prismatic parts. These four faces may have different 3-axis setups. This research uses generated 3-axis setups to make 4-axis setups. The methodology of this procedure is explained below. Figure 35 shows the block shaped prismatic parts with 3-axis and 4-axis setups.

Starting point in generating 4-axis setup is setoff 3-axis setups (fixed vector) generated by method in previous section. In addition it is observed that invariant vector for a workpiece on a rotary table is the vector of rotation axis which is orthogonal to a tool axis vector. On the other side, tool axis vector can take any direction in part
coordinates which is orthogonal to rotation axis of the table. To illustrate this, consider three setups generated by procedure in section 6 shown in Figure 35.

The methodology used to make 4-axis setups is such that, S1, S2 and S3 are the generated 3-axis setups by method described in section 6. If the part from Figure 35 is setup on 4-axis machine and tool axis vector is in the direction of setup S3 vector, in that case 4-axis setup will be the cross-product of S1 and S2 (perpendicular vector to S1 and S2) or S1 and S3 (perpendicular vector to S1 and S3) since S1, S2 and S3 will be machine in one setup (S1XS2 or S1XS3) or in opposite setup (S2XS1 or S3XS1) as explained in the introduction of this section. We have developed a rule (Figure 36) to generate 4-axis setups from generated 3-axis setups.
The rule states that when there are two different 3-axis generated setups of names \( ?\text{name1} \) and \( ?\text{name2} \) of vectors \( ?s1 \) and \( ?s2 \) respectively. Then create 4-axis setups of name \( ?\text{setup} \) whose vector is a cross-product of vectors \( ?s1 \) and \( ?s2 \).

![Figure 36 Rule for making 4-axis setups](image)

We have inserted a condition that cross-product of two 3-axis setups should not be zero since such vector doesn’t exist. We add all the features and associated processes from both 3-axis setups to generated 4-axis setup since the 3-axis setups \( ?\text{name1} \) and \( ?\text{name2} \) will be machine in just a single 4-axis setup of name \( ?\text{setup} \). Consider an example from Figure 35, 3-axis setups S1, S2 and S3 will form two identical 4-axis setups S1XS3 and S1XS2 but our rule based system randomly selects one of them. If S1XS3 4-axis setup is created that means it has features and processes from the 3-axis setup S1 and S2. In this case we add features and processes from other 3-axis setup (setup S3) which can be machined in same setup. Therefore, 4-axis setup S1XS3 now have features and processes from three 3-axis setups S1, S2 and S3. We developed the
following rule (Figure 37) to add features and processes from different 3-axis setup to existing 4-axis setup.

The rule states that if the vectors of 4-axis setup ?name1 and 3-axis setup ?name2 are orthogonal (dotproduct zero) we should then add features and processes from 3-axis setup ?name2 to 4-axis setup ?name1. In such a way the features and processes will be added from 3-axis setups to existing 4-axis setups. The following results show the 4-axis setups created for Slider part. Rule based system generates six 4-axis setups with features and processes associated with them. Out of six setups, only three setups are different and other three setups have opposite vectors of first three setups and same features and processes. Therefore, in the results Figure 38, Figure 39, Figure 40 we have shown only three different 4-axis setups.

```
(defrule add(features)(processes)(from)(remaining(setups))(to)(existing)(4-axis(setup))
  (declare (salience "Add features/processes in 4-axis setups")
? s <- (setup 4-axis (name ?name1) (vector ?vector1) (features ?features1) (processes ?processes1) (status nil))
  (setup (name ?name2) (vector ?vector2) (features ?f1) (processes ?p1))
  (test = 0 (dotproduct ?vector1 ?vector2))
  (test (not (member ?f1 ?features1)))
  (test (not (member ?p1 ?processes1)))
  =>
  (modify ?s (features (create ?features1 ?f1) (processes (create ?processes1 ?p1))))
)
```

Figure 37 Add features and processes from remaining 3-axis to existing 4-axis setups

Figure 38 shows the first 4-axis setup of vector (0 0 -1), and rule based system generates another 4-axis setup of vector (0 0 1) which is not shown since it has same features and processes. This 4-axis setup is formed with two 3-axis setups of vectors (-1 0
0) and (0 1 0) Both of these setups can machine 20 features out total 24 features of Slider part.

Figure 39 shows the second generated 4-axis setup of vector (1 0 0). Our rule based system generates another 4-axis setup of opposite vector (-1 0 0) with same features and processes, therefore only one setup of vector (1 0 0) is shown here. In this setup 20 features can be machined out of 24. This 4-axis setup is formed with two 3-axis setups of vectors (0 1 0) and (0 0 1).

```
(f-80 (MAIN::setup-4-Axis (name "setup-4-axis-Setup-TAD:-1.0:-0.0:-0.0-Setup-TAD:0.0:1.0:0.0:1") (vector 0.0 0.0 -1.0) (features "RECTANGULAR_POCKET(15)" "SIMPLE HOLE(25)" "SIMPLE HOLE(29)" "SIMPLE HOLE(26)" "SIMPLE HOLE(28)" "SIMPLE HOLE(27)" "SIMPLE HOLE(10)" "RECTANGULAR SLOT(9)" "RECTANGULAR SLOT(8)" "SIMPLE HOLE(12)" "RECTANGULAR_POCKET(3)" "SIMPLE HOLE(11)" "RECTANGULAR_POCKET(4)" "RECTANGULAR_POCKET(5)" "RECTANGULAR_POCKET(7)" "RECTANGULAR_POCKET(6)" "RECTANGULAR_POCKET(2)" "SIMPLE HOLE(31)" "RECTANGULAR_POCKET(14)" "SIMPLE HOLE(24)") (processes "end-milling-peripheral-RECTANGULAR_POCKET(15)" "drilling-SIMPLE HOLE(25)" "drilling-SIMPLE HOLE(29)" "drilling-SIMPLE HOLE(26)" "drilling-SIMPLE HOLE(28)" "drilling-SIMPLE HOLE(27)" "end-milling-slotting-RECTANGULAR SLOT(9)" "end-milling-RECTANGULAR SLOT(8)" "end-milling-RECTANGULAR_SLOT(3)" "end-milling-slotting-RECTANGULAR SLOT(9)" "drilling-SIMPLE HOLE(11)" "drilling-SIMPLE HOLE(10)" "drilling-SIMPLE HOLE(12)" "slab-milling-RECTANGULAR POCKET(4)" "slab-milling-RECTANGULAR POCKET(7)" "slab-milling-RECTANGULAR POCKET(5)" "slab-milling-RECTANGULAR POCKET(6)" "end-milling-peripheral-RECTANGULAR POCKET(2)" "drilling-SIMPLE HOLE(31)" "end-milling-peripheral-RECTANGULAR POCKET(2)"))
```

Figure 38 Generated 4-axis setup of vector (0 0 1) for Slider
Figure 40 shows the third generated 4-axis setup of vector (0 -1 0) for Slider part. The rule based system generates another setup of opposite vector (0 1 0) with same features and processes. This setup can machine 9 features out of 24 features. This 4-axis setup is formed with two 3-axis setups of vectors (0 1 0) and (0 0 1).

Therefore, total three different possible 4-axis setups are generated. The next step we performed is finalizing 4-axis setups out of three 4-axis setups for Slider.

To minimize the setups for prismatic part, it is always desired to select a setup with maximum number of features. We have applied the same reasoning here. After making all possible 4-axis setups we select only those setups which have maximum number of features.

```
f-82 (MAIN::setup-4-Axis (name "setup-4-axis-Setup-TAD:0.0:1.0:0.0-1-Setup-TAD:-0.0:-0.0:1.0") (vector 1.0 0.0 0.0) (features "SIMPLE HOLE(29)" "SIMPLE HOLE(26)" "SIMPLE HOLE(28)" "SIMPLE HOLE(27)" "SIMPLE HOLE(18)" "SIMPLE HOLE(19)" "SIMPLE HOLE(17)" "SIMPLE HOLE(16)" "RECTANGULAR SLOT(9)" "RECTANGULAR SLOT(8)" "RECTANGULAR POCKET(3)" "SIMPLE HOLE(12)" "SIMPLE HOLE(11)" "SIMPLE HOLE(10)" "RECTANGULAR POCKET(4)" "RECTANGULAR POCKET(5)" "RECTANGULAR POCKET(6)" "RECTANGULAR POCKET(7)" "RECTANGULAR POCKET(2)") (processes "drilling-SIMPLE HOLE(29)" "drilling-SIMPLE HOLE(26)" "drilling-SIMPLE HOLE(28)" "drilling-SIMPLE HOLE(27)" "drilling-SIMPLE HOLE(18)" "drilling-SIMPLE HOLE(19)" "drilling-SIMPLE HOLE(16)" "end-milling-slotting-RECTANGULAR SLOT(9)" "drilling-SIMPLE HOLE(12)" "drilling-SIMPLE HOLE(11)" "end-milling-slotting-RECTANGULAR SLOT(8)" "end-milling-peripheral-RECTANGULAR POCKET(3)" "drilling-SIMPLE HOLE(10)" "slab-milling-RECTANGULAR POCKET(4)" "end-milling-peripheral-RECTANGULAR POCKET(2)" "slab-milling-RECTANGULAR POCKET(7)" "slab-milling-RECTANGULAR POCKET(2)" "slab-milling-RECTANGULAR POCKET(5)" "slab-milling-RECTANGULAR POCKET(6)"))
```

Figure 39 Generated 4-axis setup of vector (1 0 0) for Slider
7.2 Selection of 4-Axis Setups

In case of Slider part, out of three different 4-axis setups, we choose 4-axis setup of vector (1 0 0) since it can machine 20 features out of 24. Then we look for the setup which has remaining 4 features and choose that one. 4-axis setup of vector (0 -1 0) has remaining 4 features, therefore we choose this setup.

```
(defrule select-final-4-axis-setups-from-total-possible-setups
  (declare (salience ?"finalize-4-axis-setups")
    ?s1 <-> (axis-4-sets-crossed (setup $?setups))
    (setup-4-Axis (name ?name1) (vector $?vector1) (features $?features1) (processes $?processes1))
    ?s2 <-> (setup-4-Axis (name ?name2&: (and (neq ?name2 ?name1) (not (member$ ?name2 $?setups)))(vector $?vector2)
      (features $?features2&: (and (>= (length$ $?features1) (length$ $?features2)) (neq $?features2 (create$ )))(processes $?processes2))
      (not (setup-4-Axis (name ?name3&: (and (neq ?name3 ?name2) (neq ?name3 ?name1))))
        (features $?features3&: (or (> (length$ $?features3) (length$ $?features2)) (> (length$ $?features3) (length$ $?features1))))))]
  =>
    (modify ?s1 (setup (create$ $?setups ?name2)))
    (modify ?s2 (features (complement$ $?features1 $?features2)) (status done))
    (modify ?s2 (processes (complement$ $?processes1 $?processes2))))
```

Figure 41 Finalizing 4-axis setups from total possible 4-axis setups
Finally, we have chosen only two setups out of three different setups which can machine all 24 features. We have applied this reasoning in the form of following rule.

The rule states that, when there are two different 4-axis setups ?name1 and ?name2, feature list in setup ?name1 is greater than feature list in setup ?name2 and there is no other 4-axis setup whose feature list is greater than feature list of ?name1 and ?name2 setups then remove the features from setup ?name2 which are common in setup ?name1. Similarly, processes which are common are removed.

In case of Slider, out of three different possible 4-axis setups, all features (19 features) from vector (1 0 0) setup are common in setup vector (0 0 1). Feature list of vector (0 0 1) setup is greater than vector (1 0 0) setup, therefore when the above rule gets fired, all features and processes from setup (1 0 0) are removed to reject that setup for setup planning on 4-axis mill. Similarly, third setup of vector (0 -1 0) has 9 features and 5 common features (common between (0 1 0) and (0 -1 0)) are removed to keep remaining 4 features that are not common between (0 1 0) and (0 -1 0).

Finally, we have finalized only two setups of vectors (0 0 1) with 20 features and (0 -1 0) with remaining 4 features for 4-axis horizontal mill (Figure 38 and Figure 40).

7.3 Four-Axis Setups Sequencing Based on FPN and TPN

Finalized 4-axis setups are sequenced based on the same principle as explained in section 6.5. We have developed same rule (Figure 22) to resolve 4-axis setups conflicts but we used 4-axis setup template. Following figure shows the finalized 4-axis setups for Slider part. 4-Axis setup of vector (0 0 1) has 20 features while 4-axis setup of vector (-1 0 0) has 4 features.
7.4 Operation Sequencing within the 4-Axis Setups

Once the 4-axis setups are generated and finalized operations are sequenced within the setups according to augmented FPN. Operations within the setups are sequenced in the same fashion as explained in section 0.

The figures (Figure 43 and Figure 44) are finalized 4-axis setups for Slider with sequenced operations in eligible slot (selected in bold).
Figure 43 Operation sequencing within 4-axis setup (vector - (0 0 1)) for Slider
The operation sequencing obtained for 3-axis and 4-axis mill may not be same, however our operation sequencing rules follow augmented FPN. To illustrate this we have compared operation sequencing results of 3-axis and 4-axis setups of Slider. We have shown the comparison results of only these three features – Hole 31, Hole 24 and Pocket 14. The augmented FPN among these three features are such that Hole 24 has two previous features – Hole 31 and Pocket 14. Hole 31 and Pocket 14 have no previous features (refer Figure 24 for augmented FPN of Slider).

Figure 44 Operation sequencing within 4-axis setup (vector - (-1 0 0)) for Slider

Figure 27 gives the operation sequencing in 3-axis setups of Slider. Fact f-80 shows the 3-axis setup of Slider ‘Setup-TAD:1:0:0’ and the operation sequencing in this setup is Pocket 14 → Hole 31 → Hole 24 (followed augmented FPN). Figure 43 shows the operation sequencing in 4-axis setups of Slider. Facts f-81 shows the setup ‘setup-4-Axis (name "setup-4-axis-Setup-TAD:0.0:1.0:0.0-1-Setup-TAD:-1.0:-0.0:-0.0")’ and the operation sequencing in this setup is Hole 31 → Hole 24 → Pocket 14 (followed augmented FPN). From this illustration, we can say generated operation sequencing in 3-

```
85
f-83 (MAIN::setup-4-Axis (name "setup-4-axis-Setup-TAD:0.0:-0.0:1.0-Setup-TAD:1.0:0.0:0.0") (vector -1.0 0.0 0.0) (features "SIMPLE HOLE(18)" "SIMPLE HOLE(19)" "SIMPLE HOLE(17)" "SIMPLE HOLE(16)") (processes "drilling-SIMPLE HOLE(18)" "drilling-SIMPLE HOLE(19)" "drilling-SIMPLE HOLE(17)" "drilling-SIMPLE HOLE(16)") (setup ) (next ) (previous ) (cause nil) (order 1) (status done) (eligible "SIMPLE HOLE(19)" "SIMPLE HOLE(17)" "SIMPLE HOLE(18)" "SIMPLE HOLE(16)"))
```
axis and 4-axis setup of any prismatic part may not be same; however operation sequencing follow augmented FPN.

7.5 Four-Axis Setups Precedence and Ordering

Setup precedence indicates which setup should be machined prior to which setup. 4-axis setups precedence was performed in the same fashion as explained in section 6.7.1. After the setup precedence network is generated setup ordering is performed in similar manner as explained in 6.7.2. We developed the same rule for 4-axis setup precedence and ordering but 4-axis setup template is used. The result of 4-axis setup precedence is shown in Figure 45.

We have used the same principle to order 4-axis setups as explained in section 6.7.2. The result of ordering of 4-axis setups for Slider is shown in Figure 43 and Figure 44 (underlined text). 4-axis setup with vector (-1 0 0) will be machined first and the 4-axis setup with vector (0 0 1) will be machined later. Since generated the 4-axis setups do not have setup precedence any one of them will be arbitrary machined 1st and rule based system chose 4-axis setup with vector (-1 0 0) first.

Figure 45 Four-axis setup precedence for Slider
8. FIXTURE PLANNING

This thesis research explains the fixture planning and proposes the methodology obtaining the fixture configuration (part surface for locating points). Fixture planning is the next step after setup plan is generated. Fixture planning is performed on the generated setups.

Fixture is a mechanical device to locate, hold and support workpiece during manufacturing operations. Fixture planning consists of several tasks such as fixturing configuration that involve finding locating, clamping surface points according to geometry of workpiece, conceptual fixture design, detailed fixture design and verification.

8.1 Methodology

In this section, the methodology for obtaining the part surfaces for locating points in fixture planning is explained. For prismatic parts, the general rule of 3-2-1 locating points is most commonly used. Figure 46 shows the 3-2-1 locating principle for prismatic part [29].

Figure 46 The 3-2-1 locating principle for prismatic part [29]
The part surface ‘A’ is primary datum and that has 3 locating points. The surface ‘B’ is a secondary datum that has 2 locating points. The surface ‘C’ is a tertiary datum that has one locating point.

To find out the locating surfaces for generated setups this thesis has proposed following approach. Consider Figure 47. The rectangle represents the prismatic part. It has normal vector (1 0 0) and opposite surface has normal (-1 0 0).

![Figure 47](image)

**Figure 47 Choose a part surface for 3-locating points**

First, thesis research proposes to find a part’s surface for 3-locating points. When the Setup Planning Panel has generated a ‘Setup A’ of vector (1 0 0), then 3-point locating surface will be the parallel surface, opposite of vector (1 0 0) that is the surface of vector (-1 0 0). During machining the features in setup A, the part will be rested on the
machine table on surface which has vector (-1 0 0). The following constraints should be considered while selecting face for fixturing.

- If there are multiple faces on opposite side, then the face with largest distance will be selected or the face with the maximum area will be selected. To be more precise on selecting a face from multiple faces, analysis on intermediate solid model should be performed.

- If the prismatic part has no parallel surface on opposite side, then the special purpose fixtures or different design of fixtures will be needed to hold the prismatic part securely.

Second, thesis research proposes to decide a part’s surface for 2-locating points. Since the 2-locating point surface is always perpendicular to 3-point locating surface there will be 4 perpendicular faces available for 2-locating points. This thesis research proposes to decide such a surface out of 4 for 2-locating points which has maximum surface area.

Third, when the 3-2 locating point surfaces haven been decided, then the next task is to find part surface for 1-locating point. Since the 1-locating point is always perpendicular both of 3-2 locating surfaces, this research proposes to find a face for 1-locating point which will be orthogonal to both 3-2 locating surfaces.
9. IMPLEMENTATION

This section describes the work done on how each unit of model is implemented to make the module work. The setup planning work is done in Jess (Rule Based System) [22] and remaining part of the work (Creation of facts that are needed for setup planning and visualization of setup planning results) is done in Java. Since the setup planning work has been done using rule based system, we have given it a name ‘Rule Based-Setup Planning System (RB-SPS)’. Finally this section describes the implementation of integration between rule based system and java.

9.1 Rule Based-Setup Planning System (RB-SPS)

Setup planning reasoning of this thesis research is done in Jess which is an expert system shell implemented in Java.

A rule based system is a system that uses rules to derive conclusions from conditions. It consists of an interference engine, rule base and a working memory (Figure 47). The inference engine applies the rules to working memory and control the process to obtain the output of the system. The working memory stores the data such as facts and conclusions of the rules. The rule base contains all the rules the system knows. An architecture of rule based system is shown in Figure 48 [22]. The RB-SPS system was tested initially from Jess only by running engine manually on manually created facts (prismatic part geometry and design specification) for prismatic part. Finally the RB-SPS is integrated with IMPlanner and setup planning can be done automatically. The integration of RB-SPS and IMPlanner has been explained in section 9.2.
- **Working Memory**: Working memory stores data such as facts and conclusion (new facts) derived from conditions. There are two types of facts: pure facts and shadow facts. Pure facts are defined and created entirely by Jess. Shadow facts are connected to Java objects. This research uses only pure facts. Pure facts are used to make reasoning about setup planning and integrate IMPlanner system with RB-SPS.

Each fact has a template. The template has a name and set of slots. Facts use information from templates. Facts are created automatically in java environment and passed to rule based system to fire the rules and obtain the output. Creation of facts has been explained in the section 8.3.3.

![Figure 48 An architecture of rule based system](image-url)
The main templates created for setup planning research are as follows:- feature, TAD, process, tolerance, setup, setup-4-Axis.

- **Feature template**: It stores properties (slots) of a feature such as name, type, TAD, bottom etc. ‘slot’ stores a value while ‘multislot’ stores set of values. For ex. Feature has a singular name while that feature may have several TADs. The structure of feature template is shown in Figure 49. Each slot has different types such as STRING, SYMBOL, INTEGER, FLOAT etc. that dependent on what type of data user passes from IMPlanner to RB-SPS.

```
(deftemplate feature
  "all the attributes about the feature"
  (slot name (type STRING))
  (slot type (type SYMBOL))
  (multislot TAD (type SYMBOL))
  (slot bottom (type SYMBOL))
  (multislot holeAxis (type INTEGER))
  (multislot normal (type INTEGER))
  (multislot sweep (type INTEGER))
  (multislot tolerance (default (create$)))
  (multislot nextFPN)
  (multislot previousFPN)
  (multislot nextTPN)
  (multislot previousTPN)
  (multislot next)
  (multislot previous)
  (slot lowestToleranceValue (type FLOAT)))
```

Figure 49 Feature template for setup planning

- **TAD template**: - It stores all the properties of Tool Approach Direction (TAD). The properties associated with TAD template are name, features, vector, processes, decideProcesses and decideFeatures. decideFeatures and decideProcesses are different from features and process respectively. Features and
processes store all the features and processes that are available in the current system while decideProcess and decideFeatures store clustered processes and features in a setup based on shared TAD and tolerance relations. TAD have singular name while it may have multiple features and processes. Each TAD has a 3D vector. The structure of TAD template is given in Figure 50.

- **Process template:** - It stores the properties of processes required in setup planning. It has name, type, feature and TADs. The structure of process template is given Figure 51.

```
(deftemplate TAD
  (slot name (type SYMBOL))
  (multislot features (default (create$)))
  (multislot vector)
  (multislot processes (default (create$)))
  (multislot decideProcesses (default (create$)))
  (multislot decideFeatures (default (create$)))
)
```

Figure 50 TAD template for setup planning

```
(deftemplate process
  "all the attributes about the machining operation"
  (slot name (type STRING))
  (slot type (type STRING))
  (slot feature)
  (multislot TAD (default (create$)))
)
```

Figure 51 Process template for setup planning
• **Tolerance template**:- It stores the properties of a tolerance such as name, type, tolerance value, primary datum, secondary datum and tertiary datum. The structure of tolerance template is shown Figure 52.

```
(deftemplate tolerance
  (slot name (type SYMBOL))
  (slot type (type STRING))
  (slot toleranceValue (type NUMBER))
  (slot primaryDatum (type STRING))
  (slot secondaryDatum (type STRING))
  (slot tertiaryDatum (type STRING)))
```

Figure 52 Tolerance template for setup planning

• **Setup and Setup-4-Axis templates**:- Setup template is created to store 3-axis setups information while Setup-4-Axis template is created to store 4-axis setups information. Both of these setup templates have same properties except the template name. The structure of this template is shown Figure 53.

```
(deftemplate setup
  (slot name (type SYMBOL))
  (multislot vector)
  (multislot features (default (create$)))
  (multislot processes (default (create$)))
  (slot cause)
  (multislot eligible (default (create$)))
  (slot order)
  (multislot next (default (create$)))
  (multislot previous (default (create$)))
)
```

Figure 53 Three-axis setup template for setup planning
• The **control templates** created in RB-SPS research are as follows:-
  
  o **featureList** – To save feature list required to generate TPN and augmented FPN.
  
  o **setupList** – To save 3-axis setups list required to generate 3-axis setup precedence graph.
  
  o **setupList-4-Axis** - To save 4-axis setups list required to generate 4-axis setup precedence graph.
  
  o **featuresDone** – To save the features that are clustered in setups.
  
  o **featuresMachined** – To save the features that are sequenced for machining on 3-axis mill.
  
  o **setupsMachined** – To save the setups that are ordered for machining on 3-axis mill.
  
  o **setupsOrdered-4-axis** – To save the setups that are ordered for machining on 4-axis mill.
  
  o **featuresMachined-4-axis** – To save the features that are sequenced for machining on 4-axis mill.

• **Rule Base**: - Rule engine has a rule base to store the rules. It contains all the rules that system knows. The developed rules for setup planning have been explained in section 6 and 7. The developed rules for setup planning are listed below(only major rules are listed). The total number of rules associated with each tasks is also mentioned.
  
  o Rules to identify TAD for blind and through hole feature – 6 Rules
Rules to identify TAD for blind and through pocket feature – 6 Rules
Rules to identify TAD for blind and through slot feature – 10 Rules
Rules to identify TAD for slab feature – 3 Rules
Rules for clustering features based on shared TAD and tolerance relations – 6 Rules
Rules to generate TPN and augmented FPN – 6 Rules
Rules to generate 3-axis setups – 1 Rule
Rules to resolve 3-axis setup conflicts – 7 rules
Rules to generate 3-axis setups precedence – 3 Rules
Rules to order 3-axis setups – 2 Rules
Rules to sequence operations within the 3-axis setups – 2 Rules
Rules to generate and select 4-axis setups – 4 Rules
Rules to resolve 4-axis setup conflicts – 6 rules
Rules to generate 4-axis setups precedence – 3 Rules
Rules to order 4-axis setups – 2 Rules
Rules to sequence operations within the 4-axis setups – 2 Rules

**Inference Engine:** The interference engine applies rules to data (facts) and controls the whole process to obtain the output of the system. In all rules conditions are compared to facts (by using pattern matcher) in working memory to decide which rules need to fire or activate. The rule is activated if for each condition there is a fact that satisfies the condition’s pattern. In this case unordered list of rules with the activated rules create conflict set. The conflict set
is then ordered to form agenda and process of ordering the agenda is called as conflict resolution.

The first rule on agenda is fired and entire process is repeated. Results from pattern matcher and from agenda’s conflict resolution can be preserved to perform necessary work.

9.2 Integration of IMPlanner and RB-SPS

As we have seen in section 5, that integration of IMPlanner with RB-SPS is important since RB-SPS needs features and processes information from IMPlanner. The flow of data transfer in this integration is as follows: CAD model of prismatic part is loaded to Integration Panel \(\rightarrow\) Alternate processes are generated for each manufacturing feature \(\rightarrow\) XML file is saved (has features and processes information) \(\rightarrow\) This XML file is loaded to Tolerance Definition Panel \(\rightarrow\) Tolerances are added to the feature as per design specifications \(\rightarrow\) XML file is saved (has features, processes and tolerance information) \(\rightarrow\) This XML file is loaded to Setup Planning Panel \(\rightarrow\) Run RB-SPS rules \(\rightarrow\) Display setup planning results

9.2.1 Integration Panel

In this thesis research some of the work is done in IMPlanner such as addition of tolerances and tolerance values according to part design blueprint in Tolerance Definition Panel, creation of facts (prismatic part’s data) in Setup Planning Panel and pass the facts to the rule based engine (that contains setup planning rules) to perform setup planning. The first task in the setup planning is getting features mapped of the input part and alternate processes required to manufacture these features from Integration Panel.
Following figure shows the Integration Panel [13] with features mapped and alternate processes generated for each feature.

Figure 54 Mapped features of Slider and generated processes in Integration Panel

Left panel of Integration Panel shows the mapped features and alternate processes generated for each feature (processes are visible when the feature tree in expanded). Right side of the panel shows the wireframe model of the prismatic part when it is open in Integration Panel. To generate alternate processes Integration Panel requires NX part and stock. When the NX part and stock are taken as input to Integration Panel, rbpp engine is loaded (by Load Engine button). After the engine is loaded facts are created (by Create Fact button) for given NX part. Once the facts are generated ‘rule based process selection’ (rbpp project) run behind the panel (when run engine button hit) and alternate
processes are generated for each feature (shown in the left panel). The generated alternate processes for each feature as well as feature’s information (bottom, dimensions, normal, sweep, hole axis etc.) are saved in XML (Extensible Markup Language) file and used as input for setup planning project. XML file is used in the research since it saves the data in a format that is both human-readable and machine-readable.

Figure 55 XML file saved by Integration Panel with alternate processes for Slider

The Figure 55 shows saved XML file by Integration Panel after generation of alternate processes for Slider part. Only information about feature ‘SIMPLE-HOLE (12)’ is shown in XML file such as feature dimensions (radius, holeAxis etc.), next and previous FPN features for simple hole 12 feature, two generated alternate processes required to machine simple hole 12 feature (drilling-SIMPLE HOLE(12) on ‘CncDrillSlow’ and ‘CncDrillFast’). Tool and cutting parameter information is also saved but it is not required for setup planning.
9.2.2 Tolerance Definition Panel

In this research Tolerance Definition Panel has been developed to add tolerance information for each feature. The current CAD software do not support API to extract tolerance information from CAD model, therefore we developed Tolerance Definition Panel to add tolerances externally. Tolerance information is required during clustering features into setups. Tolerance Definition Panel is basically inherited from Integration Panel. In addition GUI for entering the tolerance information has also been inherited from Ajit Wadatkar’s process selection application [30]. The Figure 56 shows the Tolerance Definition Panel.

The necessary data required for setup planning is feature dimensions, next and previous FPN features, alternate processes and tolerance information. Saving tolerance information is the next task after the XML file generated from Integration Panel. Tolerance information is saved using Tolerance Definition Panel which is explained in next section. The features and processes will be loaded when the XML file is opened. The user then can manually add tolerance information (right bottom side of the panel) for any feature. Once the tolerance information is added, XML file is saved as an input to Setup Planning Panel. Datum information cannot be saved in Tolerance Definition Panel because IMPlanner doesn’t have complete tolerance model. The method to add datum information is explained in next section. Following figure shows the saved XML file by Tolerance Definition Panel. The portion of saved XML file is shown in Figure 57 (Information about SIMPLE-HOLE(12) feature). Simple Hole (12) feature has ‘truePosition’ tolerance type with value 0.001. This information is saved and can be seen
in Figure 57 at line 12. Similarly, the tolerance information is saved for other features in the XML file.

Figure 56 Tolerance Definition Panel
9.2.3 Setup Planning Panel

The saved XML file by Tolerance Definition Panel is used as input file to Setup Planning Panel. Setup Planning Panel is also inherited from Integration Panel but we replaced the Integration Panel buttons with Setup Planning buttons except ‘Open XML File’ and ‘Save XML File’ button since the operations behind these buttons are same. Setup Planning Panel is basically developed to demonstrate performance of RB-SPS by controlling and running following tasks, create setup planning facts automatically, run setup planning rules, display TPN, augmented FPN, setup precedence graph and visualize the output (efficient setup plan). Following Figure 58 shows the Setup Planning Panel when 4-axis machine is available.
When the 4-axis machine is not available, we create setup plan for 3-axis mill. The following Figure 59 shows the Setup Planning Panel when the 4-axis machine is not available.
The difference between two panels is that we replace ‘Make 4 Axis Setups’ and ‘Show 4 Axis Setups Precedence Graph’ buttons with ‘Make 3 Axis Setups’ and ‘Show 3 Axis Setups Precedence Graph’ buttons respectively when 4-axis machine is not available.

The working of Setup Planning Panel is as follows:-

- Step 1 – Select ‘open XML File’ – Open the XML file that was saved by Tolerance Definition Panel. The primary data needed from that XML file is features list, processes, tolerances.

- Step 2 – Select ‘Load Setup Template’ – We have developed the java code behind this button such that the jess file which has all setup templates (explained in section 8.1) will be loaded into rule engine once this button is pressed. The setup templates will be available in the rule engine. Setup templates should be ready in the rule engine before facts are inserted in to rule engine since facts require templates.

```lisp
(defrule slider-datums
  (?t1 <- (tolerance (name truePosition0.001)))
  (?t2 <- (tolerance (name perpendicularity0.001)))
  (?t3 <- (tolerance (name parallelism0.001)))

  =>
  (modify ?t1 (primaryDatum "RECTANGULAR_POCKET(2)"
                      (secondaryDatum "RECTANGULAR_POCKET(4)"
                        (tertiaryDatum "RECTANGULAR_POCKET(6)")))
  (modify ?t2 (primaryDatum "RECTANGULAR_POCKET(2)"))
  (modify ?t3 (primaryDatum "RECTANGULAR_POCKET(2)"))

  (run)
  (undefrule slider-datums)
)
```

Figure 60 Datum facts creation rule for Slider
Step 3 – Select ‘Create Setup Facts’ – We have developed the java code behind this button to create feature, process, tolerance and generic facts from the available information from XML file.

We developed the rule to create datum facts for particular part and that gets fired and creates the datum facts when ‘Create Setup Facts’ button is pressed. The datum creation facts rule for Slider is shown in Figure 60.

The rule states that Slider has three tolerances shows by three jess modifiers $?t1$, $?t2$ and $?t3$. For the tolerance name ‘truePosition0.001’ rule modifies $?t1$ such that tolerance ‘truePosition0.001’ gets primary datum ‘RECTANGULAR_POCKET(2)’, secondary datum ‘RECTANGULAR_POCKET(4)’ and tertiary datum ‘RECTANGULAR_POCKET(6)’ according Slider design specification. Similarly other tolerances get their respective datum.

The datum facts creation rule for Slider is saved in separate jess file with the jess file name as ‘slider_with_slabs-tolerance-datums’. The part name given for the Slider in NX is ‘slider_with_slabs’. The datum facts creation rule file is batched and the rules from it gets fired when the ‘Create Setup Facts’ button is pressed.

We developed the java code to batch the datum facts creation fact file such that it has file name ‘part-name – tolerance-datums’. Therefore, to generate setup plan for any prismatic part, its datum facts creation rule should be saved with the file name as ‘part-name – tolerance-datums’. In that way, the desired jess file will be batched for that particular prismatic part during setup planning.
Step 4 – Select ‘Load RB-SPS Engine’ – It batches the jess file which has setup planning rules for only 3-axis mill. This research uses two jess file. One which has setup planning rules for 3-axis mill and another jess file which has all setup planning rules (for 3-axis mill and 4-axis mill). The availability of 4-axis milling machine is controlled through ‘labimp.basis.properties’ text file which is linked to setup planning project. If the property ‘SetupPlanningPanel.4_Axis_Machine_Available’ from ‘labimp.basis.properties’ is equal to ‘NO’ then ‘Load RB-SPS’ batches a file that has setup planning rules for only 3-axis milling machine. However if that property value is ‘YES’ then ‘Load Setup Rule Folder’ batches a file that has all setup planning rules.

Step 5 – Select ‘Show TPN’ – It displays the tolerance precedence network for the given prismatic part. TPN being generated using shadow facts that are connected to java object in the IMPlanner system. The TPN for Slider is already shows in Figure 17.

Step 6 – Select ‘Show Augmented FPN’ – It displays the augmented FPN for the given prismatic part. Like TPN, augmented FPN also uses shadow facts to generate graph. The augmented FPN for Slider is shown in Figure 24.

Step 7 – Select ‘Make 3 Axis Setup’ or ‘Make 4 Axis Setup’ (one of these buttons will be displayed as per the availability of 4-axis milling machine). It will cause all the rules fired since the facts are available in rule based engine and display the result (setup plan) in Setup Planning Panel. The generated setups are being considered as machine activity in IMPlanner system.
When the setup planning is completed the generated setups are sent back to IMPlanner system as machine activity and displayed to user. To view the generated setups, the user would need to click ‘Proc. plan’ tab (Figure 61). The generated setups are shown with their respective order numbers in ‘Proc. plan’ tab. The Figure 62 shows the display on 3-axis setups for Slider in Setup Planning Panel. The setup ‘Setup-TAD:0.0:1.0:0.0-1:order:6’ indicates that its machining order is 6. When the user expands the setup tree, it shows the list of processes clustered into that setup.

![Setup Planning Panel](image)

Figure 61 A pop up menu when right clicked on any process after the setup planning

- Step 8 – Select ‘Show 3 Axis Setups Precedence Graph’ or ‘Show 4 Axis Setups Precedence Graph’ (one of these button will be displayed as per the availability of 4-axis milling machine). It will display the setups precedence graph for any
prismatic part. The displayed 3-axis and 4-axis setups precedence graphs for Slider are shown in Figure 29 and Figure 45 respectively.

Figure 62 Visualization of generated 3-axis setups of Slider in Setup Planning Panel

When the 4-axis milling machine is available, the generated 4-axis setups for Slider are shown in Figure 63:-
Figure 63 Visualization of generated 4-axis setups for Slider in Setup Planning Panel
10. CASE STUDIES

10.1 Slider

The rules based setup planning system is tested on Slider part and the results of setup planning for Slider is summarized again below.

The alternate processes are generated for Slider in Integration Panel and saved it in XML. The 2D drawing of Slider is shown in Figure 65.

When the Slider CAD model is loaded into Integration Panel, rules based process selection engine will generate alternate processes for Slider features. The Figure 66 shows the generated alternate processes for Simple Hole 12 left panel of Integration Panel and on the right side of panel wireframe model is shown.

Figure 64 Slider
Figure 65 2D drawing of Slider

Figure 66 Alternate process generation for Slider in Integration Panel
When the alternate processes are generated the next step is to save the feature and processes details in XML file to be used further in Tolerance Definition Panel to add tolerances. Following figure shows the generated XML file that has features and processes for Slider part.

Figure 67 XML file saved by Integration Panel with alternate processes for Slider

This XML file is fed to Tolerance Definition Panel to add tolerances as shown in 2D drawing. Following figure shows the Tolerance Definition Panel while adding tolerances for Slider features.
Figure 68 Adding tolerances for Slider features

When the tolerances are added to Slider features XML file is saved for setup planning process. Following figure shows the saved XML file that has features, processes and tolerance information.
Figure 69 XML file saved by Tolerance Definition Panel for Slider

The next process is to feed this XML file to Setup Planning Panel to generate 3-axis and 4-axis setups. Following figure shows the generated 3-axis setups for Slider.
The generated 3-axis setups are shown geometrically below. Different color code indicates different setups. Arrows indicate TADs. Feature ‘Hole (12) (Y)’ indicates that for feature ‘Hole (12)’ (0 1 0) (positive Y axis) is chosen as a TAD and it is clustered into a setup which has got order 6 by rule based setup planning system.
Figure 71 Generated 3-axis setups and TADs for Slider shown geometrically

Following figure shows the generated 4-axis setups for Slider.
Figure 72 Generated 4-axis setups for Slider

Following figure shows the TPN for Slider.

Figure 73 TPN for Slider
Following figure shows FPN of Slider generated in IMPlanner system.

Figure 74 Feature precedence network of Slider

Following figure shows the augmented FPN of Slider (shown generated 4-axis setups in this augmented FPN).
Following figure shows the generated 3-axis setups precedence of Slider.

Following figure shows the generated 4-axis setups precedence of Slider.
Figure 77 Generated 4-axis setups precedence of Slider

Figure 78 shows the final 4-axis setups with TADs of features. Each setup is shown in different colors. The features associated with each setup have been shown in same color as that of its own setup.
Figure 78 Generated 4-axis setups and TADs for Slider shown geometrically

10.2 ANC101

The rule based setup planning for 3-axis and 4-axis machine is tested on Slider and ANC101 prismatic parts. The setup planning results are already shown while explaining every phase of setup planning. This section shows the setup planning results for ANC101 part.

Figure 79 ANC101
Figure 80 2D drawing of ANC101- side view
Figure 80 and Figure 81 shows the 2D drawing – side view and top view of ANC101 respectively. We showed only these two views that cover most of the features and tolerance information about them. The complete 2D drawing is available upon request from the author.

The ANC101 is tested in Integration Panel and rule based process selection was run for ANC101 in Integration Panel, the features are mapped and processes are selected for each feature.
Figure 82 shows the mapped features and generated processes for ANC101 in Integration Panel. The trees of ‘RECTANGULAR_POCKET(14)’ and ‘RECTANGULAR_SLOT(48)’ are expanded to show generated processes for these features. The features and generated processes are saved in XML file for further steps in setup planning. The saved XML file by Integration Panel for ANC101 has been shown below.
Figure 83 XML file saved by Integration Panel with alternate processes for ANC101

The Figure 83 shows saved XML file by Integration Panel after generation of alternate processes for ANC101 part. Only information about feature ‘RECTANGULAR POCKET (14)’ is shown in XML file such as feature dimensions (normal, bottom distance etc.), next and previous FPN features for rectangular pocket 14 feature, generated alternate processes required to machine rectangular pocket 14 feature. Since the rectangular pocket feature is actually a Slab, alternate processes such as slab milling, end-milling-peripheral processes are generated and shown in XML file. Tool and cutting parameter information is also saved but it is not required for setup planning. The necessary data required for setup planning is feature dimensions, next and previous FPN features, alternate processes and tolerance information. Saving tolerance information is
the next task after the XML file generated from Integration Panel. Tolerance information is saved into another XML file using Tolerance Definition Panel.

![Figure 84 Adding tolerances for ANC101 features](image)

The features and processes will be popped when the XML file (saved by Integration Panel) is opened. The user then can manually add tolerance information (right bottom side of the panel) for any feature. In Figure 84, profile tolerance with value 0.501 has been added to ‘RECTANGULAR SLOT(54)’ feature. Once the tolerance information is added, XML file is saved (by Save XML File button) as an input to Setup Planning Panel.
3-axis setup plan for ANC101 is shown below (Figure 85) in Setup Planning Panel. Total 9 3-axis setups are generated for ANC101 part which are shown on left panel on Setup Planning Panel. If the setups are expanded, the clustered processes will be seen.

![Figure 85 Display of generated 3-axis setups of ANC101 in Setup Planning Panel](image-url)
The Tolerance Precedence Network of ANC101 will be generated by pressing ‘Show TPN’ button. The TPN for ANC101 is shown in Figure 86.

The augmented Feature Precedence Network of ANC101 will be generated by pressing ‘Show Augmented FPN’ button. The augmented FPN with 3-axis setups are shown in Figure 87. Features are arranged according to setups. Different colors show different setups. However, same color setups indicate that setups are split due to setups conflicts.

3-axis setups precedence for ANC101 will be generated by pressing ‘Show 3 Axis Setups Precedence Graph’. Figure 88 shows generated 3-axis setups precedence graph for ANC101.
Figure 87 Generated augmented FPN with color-coded 3-axis setups

Figure 88 Three-axis setups precedence graph of ANC101
Figure 89 Generated 3-axis setups and TADs for ANC101 shown geometrically

If the 4-axis machine is available, the results of setup plan for ANC101 are as follows-
The generated 4-axis setups for ANC101 are displayed on left side of Setup Planning Panel (Figure 90). Each setup is expanded to show clustered processes in each setup. Following Figure 91 shows the augmented FPN with 4-axis setups of ANC101. Simple hole 37 initially was a part of setup 1 but it creates setup conflict with setup 2 therefore it was removed from setup 1 added into setup 1-1 to resolve the conflict. Therefore, total 3 setups were formed for ANC101 on 4-axis mill. Following Figure 92 shows the 4-axis setups precedence of ANC101. The 4-axis setup ‘Setup-TAD:0.5:0.0:0.866-Setup-TAD:0.0:0:0:1.0’ has been ordered number 1 for machining.
‘Setup-TAD:1.0:0.0:0.0-Setup-TAD:0.0:0.0’ has been given order 2 for machining and ‘Setup-TAD:0.5:0.0:0.866-Setup-TAD:0.0:0.0:1.0-3’ has been given order 3 for machining.

Figure 91 Augmented FPN of ANC101 with generated 4-axis setups
Figure 92 Four-axis setup precedence for ANC101

Figure 93 Generated 4-axis setups and TADs for ANC101 shown geometrically
10.3 Jaw-Stationary

The rule based system developed in this research is also tested on another prismatic mechanical part called as Jaw-Stationary. The CAD model for Jaw-Stationary is shown below.

![Jaw-Stationary CAD model](image_url)

**Figure 94 Jaw-Stationary**

The tolerance information of Jaw-Stationary part is obtained from its 2-D drawing (Figure 95).
Figure 95 2D drawing of Jaw-Stationary

Figure 96 Alternate process generation for Jaw-Stationary part in Integration Panel
After the generation of alternate processes, XML file is saved to feed to Tolerance Definition Panel to add tolerances as specified in Figure 95. Tolerance are added into Tolerance Definition Panel and XML file is saved to perform setup planning for Jaw-Stationary part. Below is the setup plan for vertical 3-axis mill.

![Setup Planning Panel](image)

Figure 97 Generated 3-axis setup for Jaw-Stationary

Figure 97 shows that only one 3-axis setup is generated for Jaw-stationary part in which all features can be machined respecting augmented FPN and tolerance relations. Below figure shows the TPN for Jaw-Stationary part.
The augmented FPN for Jaw-Stationary part is shown below.

Figure 98 TPN of Jaw-Stationary part

Figure 99 Augmented FPN for Jaw-Stationary part
Figure 100 Generated 3-axis setups and TADs for Jaw-Stationary shown geometrically

Since it is already discussed in section 7.1 that to generate 4-axis setups, it requires two 3-axis setups. For Jaw-Stationary our rule based system generated only one 3-axis setup therefore 4-axis setup cannot be generated in setup planning panel.
For Jaw-Stationary, setup precedence was not generated since there is only one setup for 3-axis mill and no setup generation for 4-axis mill.
11. CONCLUSIONS AND FUTURE WORK

11.1 Conclusion

This thesis has successfully developed the setup planning module (RB-SPS) to generate efficient setup plan for prismatic parts. The module is developed using a rule based approach. The rules for setup planning tasks such as clustering of features and processes, operation sequencing and setup sequencing are successfully developed. The setup planning module is successfully integrated with IMPlanner system using Java. In this research two panels ‘Tolerance Definition Panel’ and ‘Setup Planning Panel’ are developed. The Tolerance Definition Panel received XML file from Integration Panle that has features, alternate processes and next-previous features information. Tolerance information is added manually in Tolerance Definition Panel and XML file is saved. Setup Planning Panel receives XML file from Tolerance Definition Panel and generates necessary facts to perform setup planning. Moreover, Setup Planning Panel is capable to generate setup plan for 3-axis and 4-axis mill, it generates TPN and augmented FPN graphs.

The RB-SPS system is successfully tested on real mechanical prismatic parts and satisfactory results are obtained.

This thesis research proposes the methodology for fixture planning on generated setups.

11.2 Limitations

While development of RB-SPS was successful, there are still several limitations that this thesis did not accomplish:-
• This thesis research is limited to prismatic part which has only hole, slot and pocket features.
• During operations sequencing within the setups, minimum tool change criterion did not consider.
• At the time of adding tolerance information manually, datum information cannot be added manually through Tolerance Definition Panel, instead this thesis research requires a separate rule file for datum information for the particular prismatic part.
• This thesis research proposes the methodology for fixture planning for the generated setups but is doesn’t implement it.
• The system doesn’t verify the optimality of generated process plan.

11.3 Future Work
The thesis work in setup and fixture planning can be extended in several directions:

• The thesis work can be extended to generate setup plan for the prismatic part which has other manufacturing features such as chamfer, ball end, bull end slots etc.
• Currently, this thesis research is capable to generate setup plan for 3-axis and 4-axis mill. It can be extended to generate setup plan for 5-axis mill.
• During operations sequencing within the setups, tool change parameter as well as cost can be considered.
• The Tolerance Definition Panel should be extended to add datum information manually. Currently to add datum information, separate rule file is required and
that consumes time during setup planning thus it requires complete tolerance modeling in IMPlanner system.

- Currently, this thesis research proposes the methodology for fixture planning for generated setups. It can be extended to implement and generate efficient fixture planning module and integrate it with Setup Planning Module.
REFERENCES


