A
Dissertation Report
On

COMPUTER AIDED NC TOOL PATH PLANNING FOR 3-AXIS VERTICAL MILLING USING CONICAL TOOL

Submitted in partial fulfillment of the requirement for the award of degree of

MASTER OF ENGINEERING
IN
CAD/CAM & ROBOTICS

Submitted by
ABHISHEK SHARMA
Roll No. 801181006

Under guidance of

Mr. Ravinder Kumar Duvedi
Assistant Professor, MED,
Thapar University, Patiala

Mr. Sandeep Kumar Sharma
Assistant Professor, MED,
Thapar University, Patiala

MECHANICAL ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA-147004, INDIA

JULY, 2013

CERTIFICATE

This is to certify that the work done in the dissertation titled **"Computer aided NC tool path planning for 3-axis vertical milling using conical tool"**, being submitted by **Mr. Abhishek Sharma,** Registration number - 801181006, in partial fulfillment of requirement for the award of Master of Engineering degree in CAD/CAM & Robotics in the Mechanical Engineering Department of Thapar University, Patiala, is an authentic record of work carried out by me under the guidance of **Mr. Ravinder Kumar Duvedi,** Assistant Professor and **Mr. Sandeep Kumar Sharma,** Assistant Professor, Mechanical Engineering Department, Thapar University, Patiala.

Dated: 15/07/2013

(Abhishek Sharma)

(Mr. Ravinder Kumar Duvedi)
Assistant Professor, MED,
Thapar University, Patiala

(Mr. Sandeep Kumar Sharma)
Assistant Professor, MED,
Thapar University, Patiala

Counter signed by:-

(Dr. Ajay Batish)
Professor & Head of Department, MED,
Thapar University, Patiala

(Dr. S.K. Mohapatra)
Dean of Academic affairs,
Thapar University, Patiala

i

ACKNOWLEDGEMENT

I am highly grateful to the authorities of Thapar University, Patiala for providing this opportunity to carry out the thesis work. I would like to express a deep sense of gratitude and thank profusely to my thesis guides **Mr. Ravinder Kumar Duvedi and Mr. Sandeep Kumar Sharma** for their sincere & invaluable guidance, suggestions and attitude which inspired me to submit thesis report in the present form.

I am also thankful to other faculty members of Mechanical Department, TU, Patiala for their intellectual support.

My special thanks are due to my family members and friends who constantly encouraged me to complete this study.

ABHISHEK SHARMA

ii

ABSTRACT

Over the past few years, manufacturing companies have had to deal with an increasing demand for feature-rich products at low costs. Today, the availability of powerful and low cost 3D tools provides interesting opportunities to the manufacturing community, with solutions directly implementable at the core of their businesses and organizations. Mass Customization is a paradigm that produces custom products in masses. Although process planning for mass customized products is same, the tool path required to CNC machine the custom feature varies from part to part. If the tool path is created manually the economics of mass production are challenged. The only viable option is to generate the tool path automatically, furthermore, any time savings in the tool path lead to increase in profit margins.

The thesis presents a tool path planning approach for the conical end milling cutter having same taper angle to that of surface to be machined. An algorithm has been developed for 3-axis vertical milling machine using parametric equations, which has not been discussed previously for conical cutter. A zig-zag tool path foot print has been developed to attain a better surface finish on the machined part. The tool path generated is then experimentally validated by cutting a single profile with conical cutter, ball nose cutter and cylindrical end milling cutter. The resulting cavities are then tested and compared for roughness. The result validates the tool path foot print and shows that finishing achieved by using conical end milling cutter is better than the other two cutters.

ABBREVATIONS

CC	: Cutter Contact point
CL	: Cutter location point
STL	: Stereolithography
CAD	: Computer aided design
CAM	: Computer aided manufacturing
STEP	: Standard for the exchange of product model data
IGES	: Initial graphics exchange specification
ASCII	: American standard code of information interchange
CNC	: Computer numeric control
NC	: Numeric control
MCU	: Machine control unit
NURBS	: Non uniform rational B-spline
STEP	: Standard for exchange of product data
HSS	: High speed steel
APT	: Automatically programmable tool

NOMENCLATURE

u	: Parameter between initial tool position and final positions
s	: Parameter between one edge of the vertex (triangle)
t	: Parameter between other edges of the vertex (triangle)
T_1	: Initial tool position
T_1	: Final tool position
P_0, P_1, P_2	: Vertices of the triangular facet
Q_1	: Inclination of the triangular facet
Q_2	: Inclination of the edge of the triangular facet
β	: Taper angle of the tool
β_1	: Angle from radial line through the cutter tip to the bottom
R_{bottom}	: Radius at the bottom tip of the tool
R_{max}	: Radius of tool at the top
R_1	: Radii of ring circle
L	: Height of the tool
d	: Cutter diameter
R_a	: Roughness average or arithmetic average
R_t	: Maximum height of profile
R_z	: Average maximum height of profile
R_c	: Mean height of profile irregularities

LIST OF FIGURES

LIST OF TABLES

INDEX

CHAPTER 1
INTRODUCTION

1.1 NUMERIC CONTROL

Numeric control (NC) is defined as the control of the machine tool by the use of series of coded instructions consisting of numbers, letters and symbols which is understood by machine control unit (MCU). These instructions are converted into pulses of current which controls the machine's motors to carry out the required manufacturing operation on a workpiece.

The introduction of software based controls hence replaced NC hardware design with computer logic which has more control and capacity and is programmable for variety of functions at a time. As a result the modification time is reduced.

1.1.1 Advantages of CNC in manufacturing

1. The CNC program can be written, stored and executed directly on CNC machines.
2. Different CNC programs can be stored in MCU simultaneously.
3. Several machines can be attached to the main computer and program can be downloaded to any machine.
4. The CNC program can be input from flash or floppy disks or downloaded from local area networks.
5. Any portion of CNC program can be edited and playback at any time.

1.1.2 Components of CNC:

1. **Machine Control Unit**- It reads and decodes the coded instructions. It also interpolates (linear, circular, and helical) to generate axis motion commands. It controls the axis motion for driving axis mechanisms. It also controls the speed, feed for each axis and auxiliary functions like coolant or spindle ON/OFF and tool change.
2. **Machine tool**- CNC can be used to control various types of machine tools. It also has a slide table and a spindle to control its position and speed. The machine table is controlled in the X and Y axis, while the spindle runs along the Z axis.
3. **NC tool path planning**- It is divided into two types as explained.

   (1) **Manual NC tool path planning** - Manual NC tool path is done by using G and M codes. It is used where small and simple profiles are generated by manually performing calculations for finding the tool path location and position. Examples are straight cuts, hole drilling, arc cuts etc.

(2) **Computer aided NC tool path planning**- Computer aided NC tool path is helpful in generating complicated shapes, which are very difficult and time consuming using manual method. It is also very helpful, if we are working on very large workpieces and where high precision is required like in producing turbine blades.

1.1.3 Advantages of Computer aided NC tool path planning

1. All calculations are done automatically by the computer.
2. Offset calculations are done automatically.
3. Tool path location or cutter contacts (CC) locations are automatically calculated by NC tool path program algorithm.

1.2 COMPUTER AIDED NC TOOL PATH PLANNING

The computer aided NC tool path planning is widely used in today's manufacturing industries. It reduces the computational time, since most of the calculations are done by computer during the execution of program. In case of complex geometries, the computation is very difficult. Conventionally, by the use of NC tool path generation methods having continuous or triangulated free-form surfaces, two approaches were followed

(1) The generation of CL (Cutter Location) surfaces

(2) The other is the use of z-map for creating high density of discrete surface points.

The necessary task is to generate interference-free tool-paths for NC machining. The tool-paths consist of a series of gouge-free CL data. The CC points approach is a point-based method, by which CL point is found by lowering the cutter along the machining direction until the cutter contacts with the part surface. The CL surface approach creates the CL surface by inversely offsetting the tool surface along the original surface. The CL data is then computed by slicing the CL surface. All of the process is time consuming. Hence by the use of computer aided NC tool path planning, the overall machining time gets reduced. Also the validation of the program to be executed on the machine can be done by using computer.

1.3 PARAMETERS REQUIRED FOR NC TOOL PATH PLANNING

1.3.1 Tool shape / Tool parameters

The shape of the tool or the cutter shape has significant impact on the profile which is to be cut on the workpiece. The APT tool, defined below, is a generalised tool shown in figure 1.1, whose parameters can be modified to get the desired tool shapes.

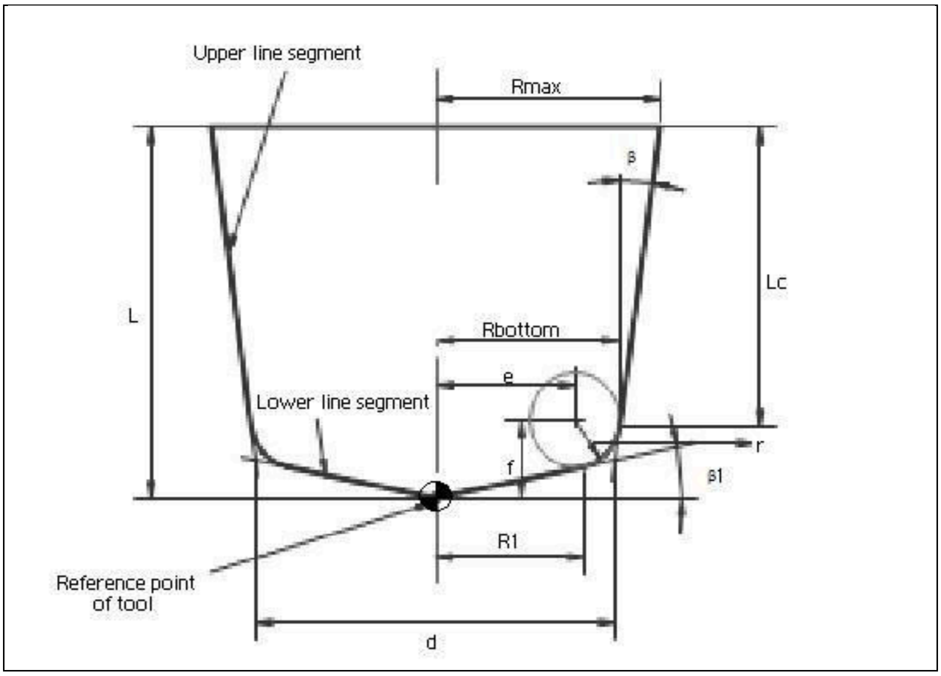

Figure 1.1: A generalised APT cutter [31]

r = Corner radius.

d = Cutter diameter, which is twice the radial distance measured from the cutter axis to the intersection of the lower and upper line segments.

e = Radial distance from the cutter axis to the centre of a corner circle; it is positive if its corner and centre are on the same side of the tool axis, otherwise it is negative.

f = Distance from the endpoint to the centre of corner circle measured parallel to the tool axis

β_1 = Angle from a radial line through the cutter tip to the cutter bottom.

β = Taper angle between the cutter side and the cutter axis; it is positive when sloping outward and negative when sloping inward from the cutter side, $0^0 \leq \beta \leq 90^0$.

L = Cutter height of the cutting edge measured along the cutter axis.

R_{max} = The maximum boundary of the cutter projecting on the part surface.

R_1 = Radii of ring circles.

Different derived shapes are listed below:-

1. **Flat end milling cutter**

   In this cutter keeping $r = e = f = \beta = \beta_1 = 0$, we get the desired shape.

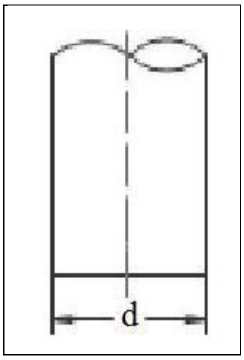

Figure 1.2: Flat End mill

2. **Ball nose end milling cutter**

In this type of cutter $e = \beta = \beta_1 = 0, d = 2r, f = r$

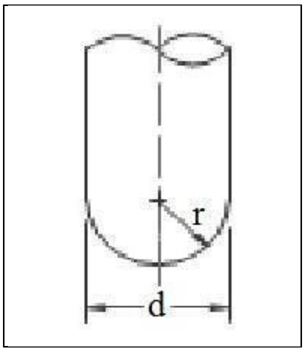

Figure 1.3: Ball nose end mill

3. **Fillet end milling cutter**

If we put $\beta = \beta_1 = 0$, $e = \frac{d}{2} - r, f = r$ below shown shape is obtained.

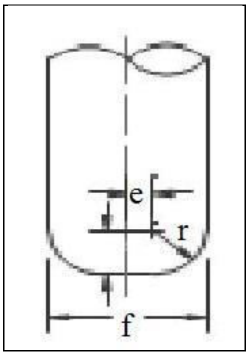

Figure 1.4: Flat end mill with fillet end

and if we put $\beta = \beta_1 = 0, e < 0, f < r, r > |e| + \frac{d}{2}$, the resulting shape is shown below.

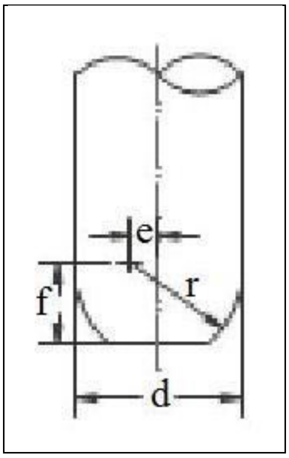

Figure 1.5 : Flat end mill with more fillet

4. Pointed end milling cutter

If $r = e = f = d = \beta_1 = 0, \beta < 0$ we get a pointed tool shape

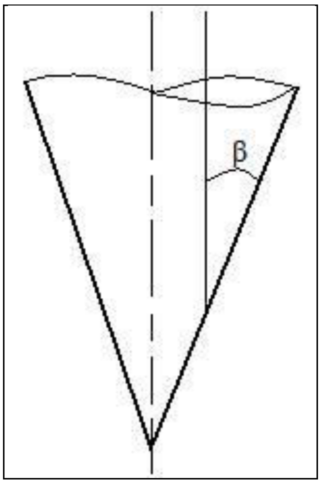

Figure 1.6: Pointed end milling cutter

5. Conical end milling cutter

It is obtained by putting $r = e = f = \beta_1 = 0, \beta < 0$

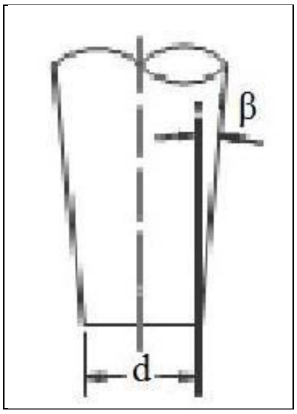

Figure 1.7: Conical milling cutter

The above described tools are the kinds of form tools which are generally used in machining operations. In case of conical tool, it provides better accessibility because of its smaller tip and stronger shank.

1.3.2 Cutter tool path planning

1.3.2.1 Raster/Zig -Zag tool path planning

It has all three translational motions. Tool penetrates the workpiece along Z-axis. The tool traverse along Y-axis till it reaches its maximum position with the provided forward feed and then moves along X-axis with the user defined side feed till the end of the part. Generally forward feed is kept 0.25 and side feed is kept 0.40 of the cutter diameter.

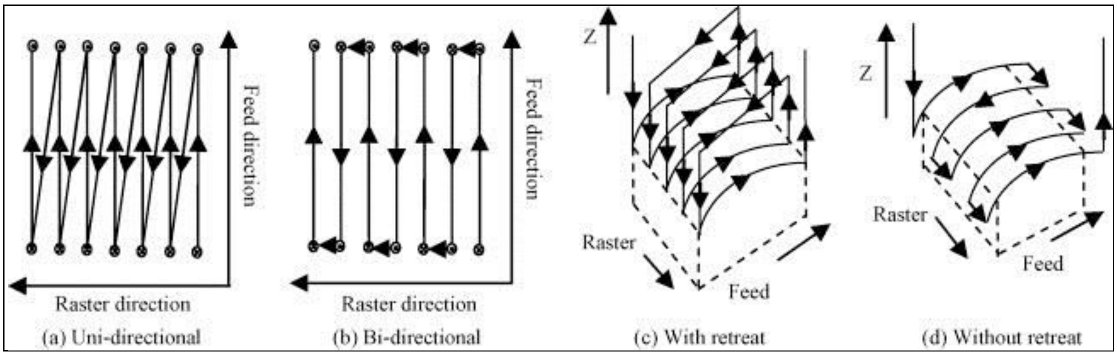

Figure 1.8: Different techniques in raster path planning [24]

1.3.2.2 Circular tool path planning-

In this strategy, Z-axis is the translational axis for tool movement. The tool moves along the X and Y-axis in a circular path. Z-axis is in positive direction where as X and Y axis can be anywhere. Circular tool path geometry can avoid discontinuities in the tool movements leading to a more consistent and smooth material removal. The feed rate and tool path radius is kept low.

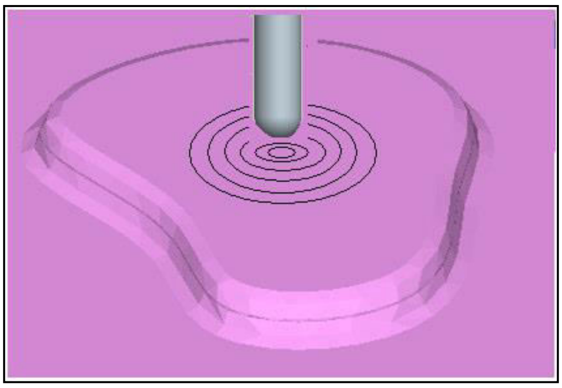

Figure 1.9: Circular tool path planning

1.3.3 Cutter contact position/ Cutter location-

The contact of the tool with the workpiece comprises of two types of paths- Cutter Contact (CC) path and Cutter Location (CL) path. The CC points are a set of points located on the sculptured surface so that the edge of the tool is scheduled to pass through, while the CL points are the set of points that the centre (or bottom of tip) of the tool is scheduled to pass through. The CL points are computed from the CC points. The interpolator in the CNC converts the CL path to motion trajectories of the entire axis in order to coordinate their motion in multi axis machining. The final obtained surface generated by CL points must be gouge free. This is done by keeping minimum scallop height by giving minimum side feed according to the scallop height.

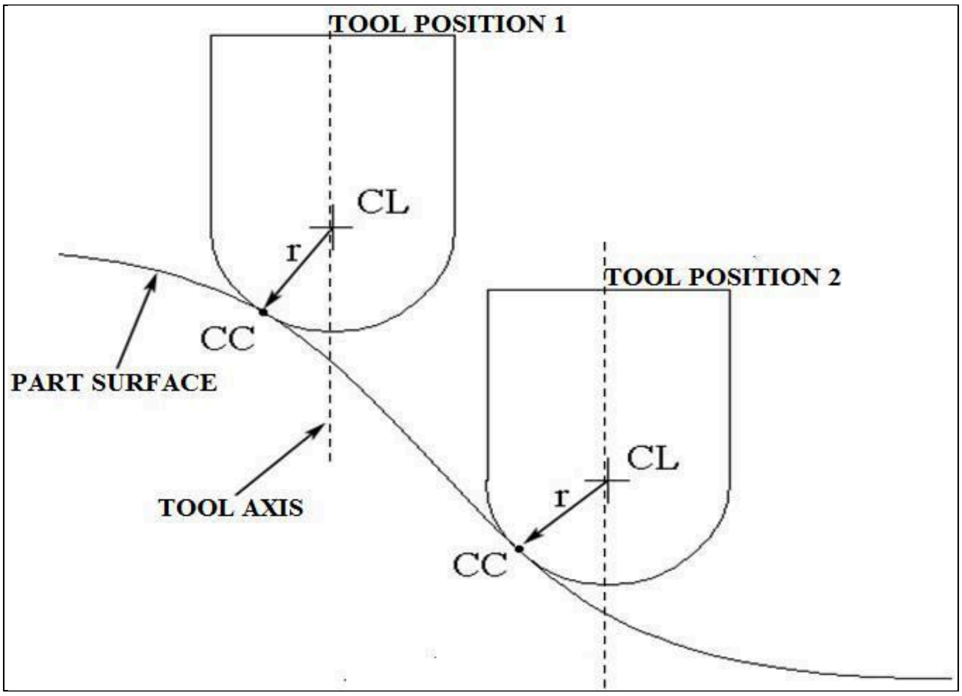

Figure 1.10: Cutter contact (CC) and cutter location(CL) points

1.3.4 Work piece geometry and topology- The surfaces of the geometry of workpiece can be made by the use of various methods as described below.

1.3.4.1 Non-Uniform Rational B-Spline (NURBS)-

NURBS are commonly used in computer-aided design, manufacturing and analysis. They can be efficiently handled by the computer programs and yet allow for easy human interaction. NURBS surfaces are functions of two parameters mapping to a surface in three-dimensional space. The shape of the surface is determined by control points. NURBS surfaces can represent simple geometrical shapes in a compact form. In general, editing NURBS curves and surfaces is highly sensitive. Control points are always either connected directly to the curve/surface or act as if they were connected by a rubber band.

The surface is usually composed of several NURBS surfaces known as patches. These patches are to be fitted together in such a way that the boundaries are invisible. A NURBS curve is defined by its order, a set of weighted control points, and a knot vector. NURBS curves and surfaces are generalizations of both B-splines and Bezier curves and surfaces, the primary difference being the weighting of the control points which makes NURBS curves rational (non-rational B-splines are a special case of rational B-splines). Whereas Bezier curves evolve into only one parametric direction, usually called s or u, NURBS surfaces evolve into two parametric directions, called s and t or u and v. NURBS forms the basis of initial graphics exchange specification (IGES) format.

NURBS curves and surfaces are useful for a number of reasons:

1. For operations like rotations and translations NURBS curves can be used by applying them to their control points.
2. They offer one common mathematical form for both standard analytical shapes (e.g., conics) and free-form shapes.
3. They provide the flexibility to design a large variety of shapes.
4. They reduce the memory consumption when storing shapes.

1.3.4.2 Bezier surface-

Bezier surface is a kind of mathematical spline used in computer graphics, design, and FEM. A Bezier surface is defined by a set of control points such that surface does not pass through the central control points but it is stretched toward each other with an attractive force. A given Bezier surface of degree (n, m) is defined by a set of $(n + 1)(m + 1)$ control points. It divides the unit square into a smooth-continuous surface. A Bezier surface will transform in the same way as its control points under all linear transformations and translations.

All $u =$ constant and $v =$constant lines in the (u, v) space of the deformed (u, v) unit square are Bezier curves. A Bezier surface will lie completely within the convex hull of its control points, and therefore also completely within the bounding box of its control points in any given Cartesian coordinate system. The points in the patch corresponding to the corners of the deformed unit square coincide with four of the control points. However, a Bezier surface does not generally pass through its other control points.

The most common use of Bezier surfaces is as nets of bi-cubic patches (where$m = n = 3$). The geometry of a single bi-cubic patch is completely defined by a set of 16 control points. These are linked up to form a B-spline surface in a similar way as Bezier curves are linked up to form a B-spline curve.

One problem with Bezier patches is that calculating their intersections with lines is difficult, making them awkward for pure ray tracing or other direct geometric techniques which do not use subdivision or successive approximation techniques. They are also difficult to combine directly with perspective projection algorithms.

1.3.4.3 B-Spline

B-spline curves share many important properties with Bezier curves, because the former is a generalization of the later. The B-spline curves have more desired properties than Bezier curves. B-spline curves require more information (i.e., the degree of the curve and a knot vector) and a more complex theory than Bezier curve. The B-spline curves satisfy all important properties that Bezier curves have. These curves provide more control flexibility than Bezier curves can achieve. In B-spline curves lower degree curves can be used and can still maintain a large number of control points. The position of a control point can be changed without globally changing the shape of the whole curve. They have finer shape control. But B-spline curves are still polynomial curves and polynomial curves cannot represent many useful simple curves such as circles and ellipses.

Different formats for saving these surfaces are:-

1.3.4.4 STEP

Standard for the Exchange of Product Data (STEP) is ISO standard industrial automation systems product data representation and exchange format. The file structure for a STEP file has a modular structure which makes it easier for developers to adapt the format to their own needs. Due to its ASCII structure it is easy to read with typically one instance per line. The file extensions .stp and .step indicates that the file contain data conforming to STEP Application Protocols. The file is split into two sections- header and data.

Header section

The HEADER section has a fixed structure consisting of 3 to 6 groups in the given order. Except for the data fields timestamp and FILE SCHEMA all fields may contain empty strings.

FILE_DESCRIPTION- The version and conformance option of this file.

FILE_NAME- Name of this exchange structure. It may correspond to the name of the file in a file system or reflect data in this file. There is no strict rule how to use this field.

FILE_SCHEMA- Specifies one or several Express schema governing the information in the data section. The last three header groups are only valid in second edition files.

FILE_POPULATION- Indicates a valid population (set of entity instances). This is done by collecting data from several data sections and referenced instances from other data sections.

SECTION_LANGUAGE- It allows assignment of a default language for either all or a specific data section.

SECTION_CONTEXT provide the capability to specify additional context information for all or single data sections.

Data section

The Data section contains application data according to one specific express schema. The encoding of this data follows some simple principles.

Instance name: Every entity instance in the exchange structure is given a unique name in the form "#1234".

Mapping of attribute values- Only explicit attributes get mapped. Inverse, Derived and re-declared attributes are not listed since their values can be deduced from the other ones.

Mapping of other data types- Enumeration, Boolean and logical values are given in capital letters with a leading and trailing dot such as ".TRUE.".

1.3.4.5 IGES

The Initial Graphics Exchange Specification (IGES) is a file format that defines a neutral data format which allows the digital exchange of information among Computer-aided design (CAD) systems. The IGES file is composed of 80-character ASCII records. Text strings are represented in "Hollerith" format, the number of characters in the string, followed by the letter "H", followed by the text string, e.g., "4HSLOT" (this is the text string format used in early versions of the FORTRAN language). It represents a slot, with the points at the centres of the two half-circles that form the ends of the slot, and the two lines that form the sides.

The file is divided into 5 sections: Start, Global, Directory Entry, Parameter Data, and Terminate indicated by the characters S, G, D, P, or T. The characteristics and geometric information for an entity is split between two sections-(1.) in a two record, fixed-length format (the Directory Entry, or DE Section) and (2.) in a multiple record, comma delimited format (the Parameter Data, or PD Section), as can be seen in a more human-readable representation of the file. One of the unique features of the IGES standard is that it was the first ANSI Standard to be documented using itself.

1.3.4.6 STL

STL (Stereolithography) format is rather conceptually simple and sufficiently accessible as it repetitively describes every normal and vertex of triangular facets built for object approximation. Facet data is saved in computers in two types of data format: text (ASCII) format and binary format. The text STL format is a set of facet descriptions in the form of ASCII containing its unit normal vector and 3D coordinates of three vertices. An STL file in text format is obviously redundant for computer storage as it records every character and digit of items. Although the content of a text STL file is readable, its file size is so large that it is generally used as a testing tool. The structure of text STL format is as follows:

solid solid_name
<facet list>
facet normal $\left(N_x, N_y, N_z\right)$
outer loop
vertex $X1, Y1, Z1$
vertex $X2, Y2, Z2$
vertex $X3, Y3, Z3$
endloop
endfacet
endsolid

Where $\left(N_x, N_y, N_z\right)$ is a normal vector, and $(X1, Y1, Z1), (X2, Y2, Z2)$ and $(X3, Y3, Z3)$ are coordinates of vertices [15]. Unlike text STL format, binary STL format is more compact and therefore more efficient for data processing because vectors and coordinates are saved as floating-point numbers, each of which occupies four bytes of computer memory. The binary STL format contains file head, facet number and facet list. STL is widely accepted by most commercial CAD software and rapid prototyping equipment due to the obvious advantage of its topologically simple and robust nature. STL format is composed of only one type of element, a triangular facet, which is defined by its normal and three vertices. All the triangular facets described in a STL format file constitute a triangular mesh to approximate modelling surfaces.

However, drawbacks exist in STL format. Flaws may appear in the process of creating triangular facets, although they can be checked and corrected afterwards. Incorrect normal and inconsistent normal are two cases of inconsistency problems in STL. The former problem happens when the facet normal generated by the CAD system is different from that calculated from facet vertices, and the latter is owing to inconsistent orientation of the normal of adjacent triangular facets. Another kind of flaw is malformation, for instance, once an STL triangular facet is too thin to keep its triangular shape, it may collapse to be a gap, crack or hole. Third problem is illegal overlapping. When a facet vertex is located on the edge of another facet or when two facets intersect with each other, the two facets are partly overlapping, which breaks the STL rule that each triangular facet must share two vertices with every adjacent facet [5].

In addition, STL has some disadvantages in both its format and applications. Redundant depiction of geometric elements in STL format, which means each vertex of a triangular facet is recorded at least four times brings extra computational memory occupation and time consumption. Another shortcoming is that STL file size is not the same with its approximation accuracy when the required approximation accuracy and pronounced curvature of an object surface increases in case of complex surface and, the size of the generated STL file is dramatically enlarged. Finally, STL format records only the geometry of object surface and lacks object attributes

.

1.3.5 Types of Machining

Milling is defined as the use of rotary cutters to remove material from a workpiece in a direction at an angle with the axis of the tool. It covers a wide variety of different operations and machines, on scales from small individual parts to large. It is one of the most commonly used processes in industry and machine shops today for machining parts.

End Milling

End milling is done with the help of end mills. End mills are those tools which have cutting teeth at bottom, as well as on the sides. They are usually made from high speed steel (HSS) or carbide, and have one or more flutes. They are the most commonly used tools in a vertical mill.

Flank Milling

In flank milling the material removal is done with the side of the cutter in comparison to point milling where the cutting is done with the bottom edge of the cutter. It shapes the design surface based on the trajectory it follows and the shape of the tool. It also increases the metal removal rate, lowers the cutting forces and ensures improved accuracy.

Sculptured surface machining

This kind of machining is relatively difficult to machine due to complex geometry. It is mainly done in two steps-roughing and finishing. It is done by using two approaches-CC surface approach and CL approach. In CC approach CC points are converted into CL points where as in CL approach CL surface is used as path generation surface. Hence CL approach is better than CC approach [18].

Categorization of milling operations on the basis of tool orientation

1. Vertical milling

In the vertical mill the spindle axis is vertically oriented. Milling cutters are held in the spindle and rotate on its axis. The spindle can generally be extended, allowing plunge cuts and drilling. There are two subcategories of vertical mills: the bed mill and the turret mill.

A turret mill has a stationary spindle and the table is moved both perpendicular and parallel to the spindle axis to accomplish cutting.

In the bed mill, however, the table moves only perpendicular to the spindle's axis, while the spindle itself moves parallel to its own axis.

Turret mills are generally considered by some to be more versatile of the two designs. However, turret mills are only practical as long as the machine remains relatively small. As machine size increases, moving the knee up and down require considerable effort and it also becomes difficult to reach the quill feed handle (if equipped). Therefore, larger milling machines are usually of the bed type.

2. Horizontal milling

A horizontal mill consists of x–y table, but the cutters are mounted on a horizontal arbor across the table. The cutters have good support from the arbor and have a larger cross-sectional area which enables quite heavy cuts for rapid material removal rates. These are used to mill grooves and slots. Plain mills are used to shape flat surfaces. Several cutters can be used together on the arbor to mill a complex shape of slots and planes. It is also easier to cut gears on a horizontal mill. Some horizontal milling machines are equipped with a power-take-off provision on the table. This allows the table feed to be synchronized to a rotary fixture, enabling the milling of spiral features such as hypoid gears.

Milling can also be categorised according to surface finish

1. **Roughing Milling:** The workpiece material is removed leaving a semi-finishing allowance. There is no emphasis on workpiece accuracy. An end mill is generally used because of its high metal removal rate and longer life.
2. **Semi-finish Milling:** It is done after roughing operation to match the exact profile to be obtained. The remaining material is then removed leaving a finishing allowance of uniform thickness. The process is generally performed using ball end mill.
3. **Finishing Milling:** It is done to remove any material left on the corners or on the profile to be machined. In this process, the final shape of the part is achieved using a conical cutter. Since high precision is required in this process, hence it is quite time consuming due to low depth of cut.
4. **Pencil cut milling:** In sculptured surface machining, the milling process usually consists of several consecutive stages: roughing, semi-finishing, finishing and clean-up. After the finishing process, the clean-up machining removes the remaining material left at the sharp corners or edges. Clean-up machining can be classified into two major types: pencil-cut and fillet-cut. In case of pencil-cutting, usually uses a single tool path to clean up the corner regions. On the other hand, fillet-cutting generally uses multiple cutting passes to clean up the corner regions with a relatively smaller cutter. The major benefit of pencil-cutting includes better surface finish and shorter total machining time for complex part surface machining.

1.4 FOLLOWING INPUTS ARE REQUIRED FOR COMPUTER AIDED NC PART PROGRAMMING ALGORITHM

1. A 3-dimensional part model is generated in CAD software like Pro E, and then it is converted into STL format with file having .stl extension.
2. Tool path foot print strategy is required to get data about tool forward feed and side feed.
3. Tool types and different parameters used to define the required tool.
4. The required depth of cut needed to generate the profile on the work piece.

1.5 NC TOOL PATH VERIFICATION

The tool path verification can be done by the use of simulation. The simulation has recently become essential, which means to execute the programs in complete safety in an unmanned mode to guarantee a high quality of machining (product accuracy and machining efficiency). This makes it possible to reduce the product development lead times and costs by correcting all the potential errors before the real machining of the part. It can be done in following ways [22].

(a) **Special environment:** A z-buffer approach has been used in continuous removal of material in a virtual environment. The simulation consisted of determining of volume of tool material removed from the part by the tool during a tool path. The toolpath must be discretized into elementary positions representing different positions of the tool centre.

(b) **CAD software:** Simulation can also be done by the use of different CAD software available. They use Boolean operations which are pre-programmed in the software.

(c) **Commercial software:** Some special software are developed only for the simulation purpose. The software does three types of visual simulations.

1. The visual simulation is used to obtain visual images of the machined part.
2. The geometrical simulation is used to check the absence of collisions, the tool engagement, the choice of effective cutting speed and the type of machining.
3. The physical simulation is used to evaluate further the cutting forces and the thermal distributions among others.

(d) Actual Machining: The above methods do not include the actual forces and chatters produced on the machine. Thus first part is always produced as a test part by actual machining to validate the simulation done on the workpiece.

1.6 PRESENT WORK

A lot of work has been done in the field of machining since the industrial revolution. As the surfaces are becoming more complicated and due to development in NC technology, the need of the milling operations is growing with each passing day in manufacturing industry. The major part of the development has been done for ball end milling cutter and cylindrical cutters. In case of conical end mill for 3-axis machine, most of the work has been done for flank milling using generalised equations based upon APT cutter. But no attempt has been made for development of generalised mathematical formulation for finding cutter location for conical tool for STL surfaces. Moreover conical end mill is used because of its accessibility and rigid shank which results in better finish and also it reduces the cutting time for machining of surfaces inclined at inclination of conical tool.

CHAPTER 2
LITERATURE REVIEW

A lot of research has been done in the field of tool path planning. The literature review has been divided into following parts.

2.1 TOOL SHAPES

The tool shape is a very important factor in machining operations. Radzevich [28] has discussed various kinds of tools available for cutting operations. Chih Ching Lo [7] has discussed the behaviour of ball end milling cutter. In order to improve the machining efficiency, using this cutter, he used a cutter of larger diameter and skipped the region of cutter interference. The skipped region then can be machined by using a small diameter cutter. Hence it is a time consuming process since two cutters are used. LiMin Zhu *et. al.* [34] discussed cylindrical and conical cutters used in flank milling. In this paper he discussed, that both the cutters are good for cutting, but since conical tool has better accessibility and rigid shank it is better than cylindrical tool for complex geometries. Hence we concluded that conical tool is better than all tool in order to increase the machining rate and accuracy.

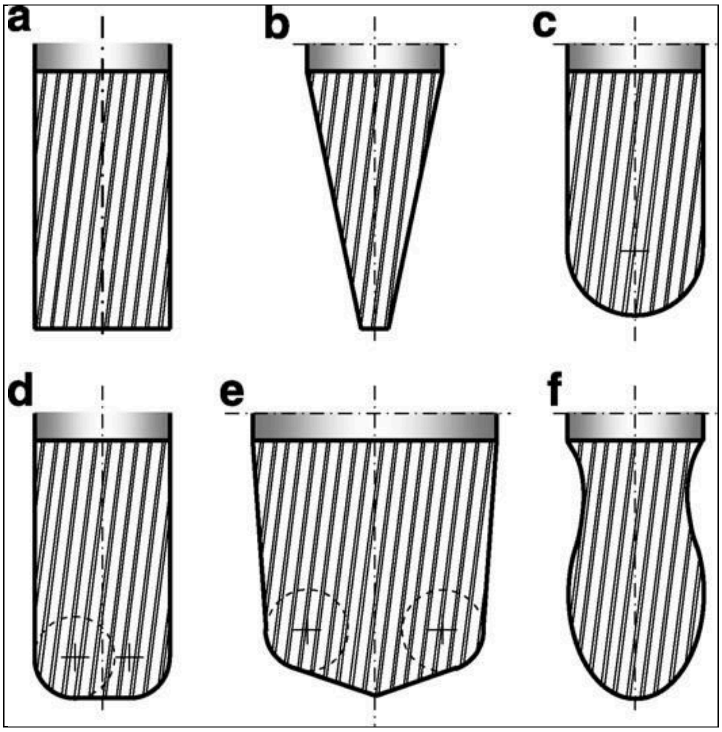

Figure 2.1: Various cutting tools for machining a given sculptured surface P on multi-axis NC machine(a) Cylindrical (b) Conical (c) Ball-End (d) Fillet-End (e) APT tool (f) General Cutting Tool

2.2 INPUT PART GEOMETRY FOR COMPUTER AIDED NC PART PROGRAMMING

In case of computer aided NC programming for a 3-axis vertical milling machine various cutters can be used. The CL surface produced can be stored in various formats. Kim and Yang [25] in there paper discussed about the triangular mesh generation for an APT cutter using NURBS. In the paper, they discussed that the triangular mesh based NC tool path generation approach is numerically more stable than surface-based tool path generation to produce a gouge free surface and also for pencil cut and remaining cutting processes. Langeron *et. al.*[30] discussed about the IGES, STEP formats used during different stages of the NC tool path programming using NURBS. Yau *et.al.* [31] discussed about the STL and IGES formats. The paper concludes that STL is better than IGES since STL is easy to implement because IGES format contains more design related information. Nagy and Matyasi [15] reviewed about techniques in discrete surface modelling which are needed in rapid prototyping, such as management of STL files, analysis of a triangle mesh in STL format to detect and to heal topological defects, construction of appropriate data structures for shape description, and smoothing of meshes. Chuang and Yau [26] discussed about STL in z-level contour machining.

2.3 TOOL PATH WITH PENCIL/FINISH CUT MACHINING

In some of the latest commercial CAM software, the pencil-cut function is provided for machining parametric surfaces or triangulated surfaces. However, most of the commercial CAM systems suffer from one or more of the following problems:

1. Fails to correctly identify the pencil-cut curves that are obvious to the human user.
2. Fails to generate pencil-cut tool paths correctly.
3. The software crashes during the process of generating the pencil-cut tool path of complex surfaces.

Ren *et.al.* [32] discussed about technique of pencil-cut curve refinement to overcome the limitation due to the discrete CL-net intervals, and the smooth pencil-cut paths are made complete at sharp corners. Hence it is imperative to develop a more robust and accurate method for pencil-cut machining. To automatically generate accurate and efficient pencil-cut paths, the following major issues must be addressed:

1. Identifying the pencil-cut points on a part surface.
2. Tracing the pencil-cut points to construct the continuous pencil-cut curves.
3. Preventing pencil-cut tool paths from intertwining in the tracing process.

For complex part surfaces, all these issues are quite challenging for generating good pencil-cut tool paths. Zhu *et.al.* [34] discussed about pencil-cut machining having the input of CAD models in form of triangulated surface models, which are commonly referred to as the STL file format. The input surface models are defined as the STL models, and the notation of pencil-cut curve is interchangeable with pencil-cut tool path. A new Material-Side-Tracing (MST) method is discussed and the pencil-cut curve refinement techniques are proposed for 3-axis pencil-cut path generation.

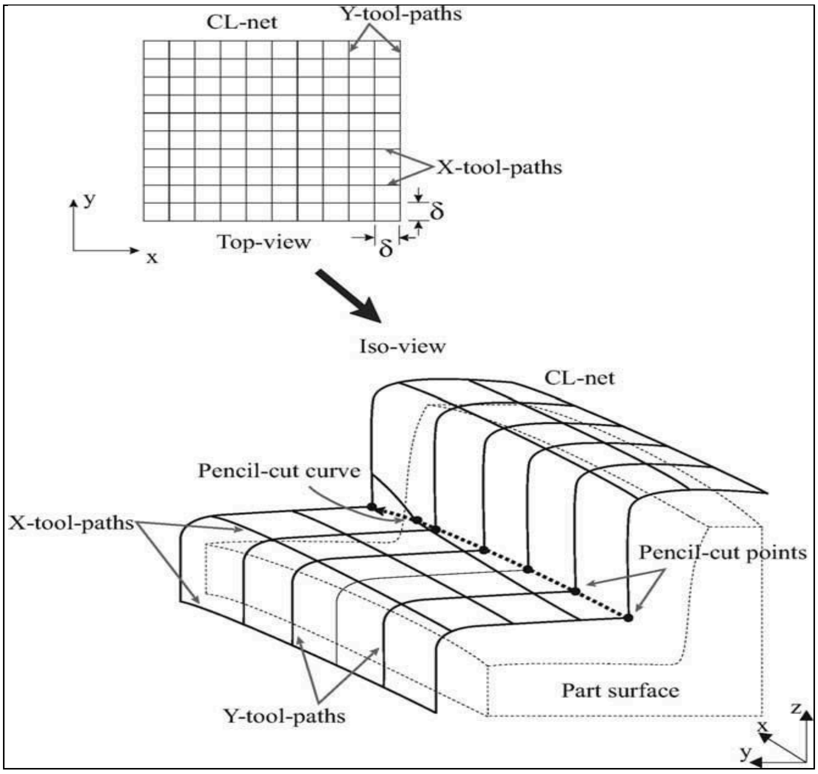

Figure 2.2: Pencil cut machining points for finishing

2.4 TOOL POSITIONING FOR CONICAL TOOL

As the use of conical cutter is increasing, so lots of work has been done in improving the tool position. Monies *et. al.* [8,9] first presented a conical tool based milling strategy in which the conical tool has three tangent points: two are on the directrices curves and one is on the ruling line between the directrix curves. The only hypothesis is that the curvatures of the directrix curves in the zone considered are kept constant. This method lead to seven equations and the solution of these equations gives the final tool position. In another work [10] they proposed an algorithm to calculate the interference error between the cutter and the workpiece so that one can determine the optimal cutter dimensions (cone radius and angle). Chiou [19] described a swept envelope-based method for tool positioning. The initial cutter positions were located to contact with two directrices. Then the swept profile of the cutter was calculated based on the cutter motion. Finally, the cutter locations were adjusted to reduce the machining errors by comparing the swept profile with the design surface. All the above methods focused on optimizing the tool position. Chaves *et al.* [33] proposed a novel approach that optimizes the tool shape to reduce the interferences. Zhu *et.al.* [34] discussed about the optimum position of the conical tool for approximating the tool envelope. In the paper is presented a method to efficiently compute the signed distance between a point in space and the swept surface without constructing the swept surface itself. Also discussed a three-step optimization approach, which needs to solve three equations numerically. The program is easy to implement and fast to compute.

2.5 CONCLUSION OF LITERATURE REVIEW

The literature related to Computer Aided NC tool path planning strategies and tool have been discussed elaborately in the previous sections. As apparent from the literature review, STL format has been proved to be the best format for NC tool path planning since it reduces complications as compared to other formats. The research for sculptured surface machining using conical tool has been very less till now. But the use of conical tool has been proved beneficial for sculptured surface machining because of its accessibility and rigid shank also it increases the machining time. Also the zig-zag path foot-print has been proved to be better for increasing the machining accuracy.

CHAPTER 3
PROBLEM DEFINITION AND MATHEMATICAL MODEL

3.1 PROBLEM DEFINITION

A generalized tool path generation algorithm for 3-axis vertical milling machine with conical end mill cutter has been developed in Microsoft Visual C++ 2010, which moved in the zig-zag path foot print with uniform forward and side feed for roughing and finishing passes, to develop a gouge free surface with the help of .STL (Stereolithography) file as input of the algorithm which gives all the information about the work piece. By taking the STL (Stereolithography) file as initial input from the CAD modeller, the computer program generates the cutter location (CL) points in a text file which is to be used in the algorithm described below.

3.2 MODELLING OF TOOL PATH GENERATION ALGORITHM

The NC tool path planning strategy for computer aided NC 3-axis milling machine has been discussed in this section in detail.

3.2.1 Inputs required for NC tool path planning

The various inputs which are used for the machining to be done are:

1. STL data for CAD model: The STL file is used as input to the tool path generation algorithm. This STL can be made from any of the CAD model. Every CAD software has capability to store the CAD model as .stl format. For now , Pro-E has been used to make the .stl file. This discretizes the CAD model in small number of triangles name facets. Also among the two formats available for STL storage, ASCII format has been used.

2. Tool parameters: The tool tip radius, taper angle and tool length defines the profile of the tool. Hence the parameters of the tool are to be decided.

3. Depth of cut: The other input is depth of cut for the pass. The workpiece generally is of much bigger size then the part dimensions to be manufactured. Hence proper depth of cut is to be kept so that tool might not break.

4. Finishing allowance: The another value which user inputs is the finishing allowance which the roughing tool should leave for finishing tool. Generally it is kept 2mm but the user may enter another value if the part is not so complex or the tool used for finishing is of higher radius.

5. Forward feed: The forward feed is usually kept 0.4 times diameter of the cutter.

6. Side feed: The value of side feed is to be entered by the user. This value is kept 0.25 times diameter of tool for roughing pass and even smaller for finished passes.

The STL file which is used as the input for the tool path algorithm contains numerals, strings which are not required. Thus there is a need to refine the STL file. A computer algorithm is used for this purpose. It automatically ignores the unnecessary data and strings which are not desired in the coming program. The STL file contains the vector normal and coordinates vertices of all facets.

3.3 MATHEMATICAL MODEL

The entire surface to be machined is represented mathematically by the vertices of triangle known as facets and surface normal called facet normal. The tool path planning algorithm has been designed to check if conical end milling cutter is touching with the part of the triangulated surface. The parameter u is set for the extreme position of the tool, and it changes along the surface as tool moves along a raster path in X-axis and Y-axis with suitable side feed and forward feed. The tool surface may touch the triangle on edge, on vertex or inside the triangle. The tool positioning strategy must stop the tool forward motion at first contact of the tool with either of triangle surface, edge or vertex for which following checks have been applied to extract the information about tool position along Z-axis of the 3-axis NC Vertical milling machine.

The conical cutting tool used has to move on the NC tool path touching the triangles, edges and vertices. The workpiece can touch tool profile at two contact regions:

1. Flat bottom part of tool.
2. Conical/taper surface of the tool.

The tool axis $\hat{T}$ will always remain parallel to z-axis, with tool reference point denoted by $T_1(0,0,1)$.Hence we come across with three types of checks to check intersection of any triangle, edge or a vertex with the tool profile. Tool parameters L, β, R_{bottom} defines the tool and let T_c be cutter contact point on the axis of tool.

Also, $R_{max} = R_{bottom} + L\tan(\beta)$

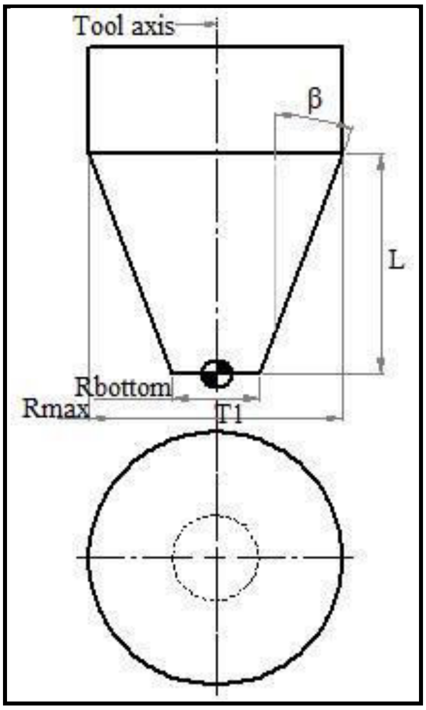

Figure 3.1: The conical tool

3.3.1 Triangle check

This check is performed to find out the intersection of the tool with the triangular facet with vertices P_1, P_2, P_3 and u is the parameter from T_1 to T_2. Initially, slope of the triangular facet is calculated using formula given below to find the orientation of the triangle with respect to tool axis.

$$\theta_1 = \tan^{-1}\left[\frac{N_z}{\sqrt{{N_x}^2+{N_y}^2}}\right]$$

Let $\hat{n}$ is the surface normal of the triangular facet having components (N_x, N_y, N_z) and $\hat{T}$ is the vector along the axis of tool, where $T = T_2 - T_1$.

In such case, we use the following equation to find the tool location:

$T_1 + u(T_2 - T_1) + \hat{n}_p.R_{bottom} = P_1 + s.(P_2 - P_1) + t(P_3 - p_1)$, where$0 \leq u, s, t, s + t \leq 1$
It can be solved by using matrix shown on next page.

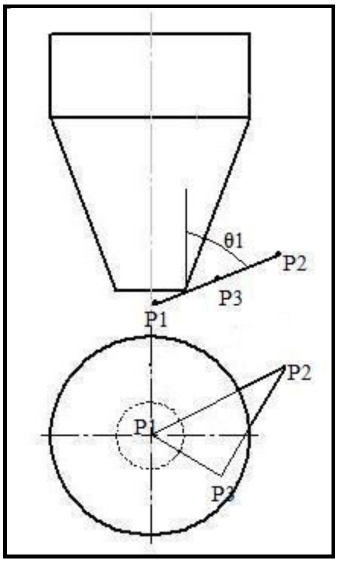

Figure 3.2: The inclination of θ_1 with facet and tool axis is shown

$$\begin{bmatrix}(T_2-T_1)_x & (P_1-P_2)_x & (P_1-P_3)_x \\ (T_2-T_1)_y & (P_1-P_2)_y & (P_1-P_3)_y \\ (T_2-T_1)_z & (P_1-P_2)_z & (P_1-P_3)_z\end{bmatrix}\begin{bmatrix}u\\s\\t\end{bmatrix}=\begin{bmatrix}\left(P_1-T_1+\hat{n}_p.R_{bottom}\right)_x\\ \left(P_1-T_1+\hat{n}_p.R_{bottom}\right)_y\\ \left(P_1-T_1+\hat{n}_p.R_{bottom}\right)_z\end{bmatrix}$$

3.3.2 Edge check

This check is performed to see if the tool moving along the line will touch any of the three edges of the triangle. If tool is touching an edge, it must satisfy the boundary conditions.

Hence we get following cases, let edge is$(P_2 - P_1)$.

Case 1: When edge is perpendicular to tool axis

In this case, $(P_2 - P_1).\hat{T} = 0$. Thus firstly, distance from tool axis is found by equation written below.

$d_1 = \frac{(P_2-P_1)\times\hat{T}}{|(P_2-P_1).\hat{T}|}.(P_1 - T_1)$. Based on value of d_1, two sub-cases can be found.

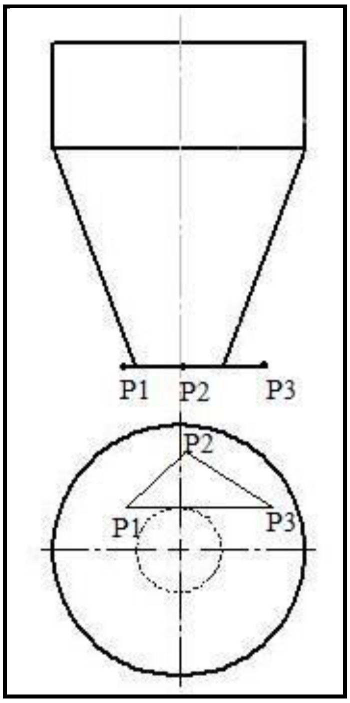

Figure 3.3: Facet position when edge is perpendicular to tool axis

a) If $0 \leq d_1 \leq R_{bottom}$,then value of $u = \frac{(P_1-T).T_1}{|T_2-T_1|}$.

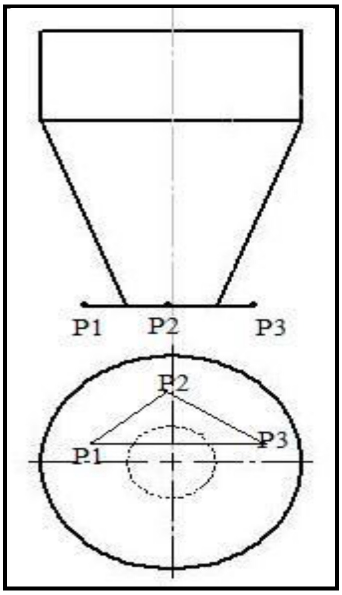

Figure 3.4: Facet position when edge is perpendicular and lies at flat end of tool

b) If $R_{bottom} \leq d_1 \leq R_{max}$, then $u = \frac{(P_1-T).T_1-L_1}{|T_2-T_1|}$, where $L_1 = \frac{(d_1-R_{bottom})}{\tan(\beta)}$.

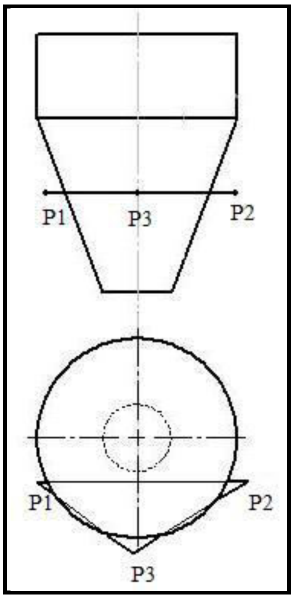

Figure 3.5: Facet position when edge is perpendicular and lies between R_{bottom} and R_{max}

Case 2: When tool is inclined with respect to the taper of conical tool

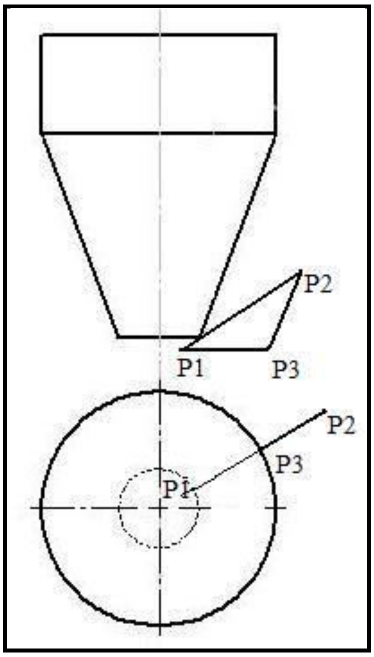

Figure 3.6: Facet position when tool is inclined away from the tool axis

In this case the tool will intersect edge at the flat end of the tool. It can be solved by using equations with conditions for (u, s) parameters. The equations are written below along with condition.

$|T_c - P| = R_{bottom}$ and $(T_c - P).T = 0$, where $o \leq (u, s) \leq 1$

3.3.3 Vertex Check

When we are talking about vertex check, it is quite easy to find u value. Let P_1 be the vertex. The distance from the vertex is calculated using relation,

$d_2 = |\{P_1 - T_1\}.T - P_1|$. Now we have two cases, corresponding value of d_2.

Case 1: If $0 \leq d_2 \leq R_{bottom}$, then

Hence, $u = \frac{(P_1 - T_1).T_1}{|T_c - T_1|}$.

Case 2: If $R_{bottom} \leq d_4 \leq R_{max}$, then

$u = \frac{(P_1 - T_1).T_1 - L_1}{|T_c - T_1|}$, where $L_1 = \frac{(d_2 - R_{bottom})}{\tan(\beta)}$.

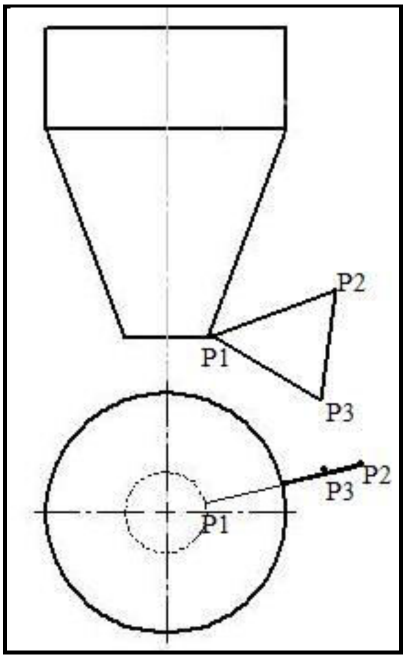

Figure 3.7: Facet position when vertex is touching the tool at a single point

3.4 SOLUTION PROCEDUTE

A computer program in Microsoft Visual C++ 2010 has been developed for the mathematical procedure defined in sections 3.2 and 3.3 above. The solution procedure for NC tool path generation for 3-axis NC vertical milling lathe has been discussed stepwise below:

i. The STL file is generated in Pro-E modelling software.
ii. The STL file is attached with the program and then the program is run on the computer.

iii. The input STL data for the 3-Dimensional model of the part is used to extract the required dimensions of the part for the algorithm.

iv. The other inputs which are required during the program are also asked from the user like, depth of cut for roughing pass, finishing allowance. After these inputs, the program starts running.

v. The maximum size of the work piece is calculated to get the stock size.

vi. The program runs for the whole of the work piece. It checks the extreme tool position where the value of u is least at every point and after storing that value, it moves to next position.

vii. Now the program applies the roughing passes moving in a zig-zag path having some forward feed in y-axis.

viii. As tool moves to maximum value of y, side step feed is given along x-axis to finish the operation till it reaches maximum value of x.

ix. After applying the roughing passes the tool goes for finishing pass.

x. All Cutter Location (CL) points are saved in auto generated text files.

The flow chart for methodology is shown below:

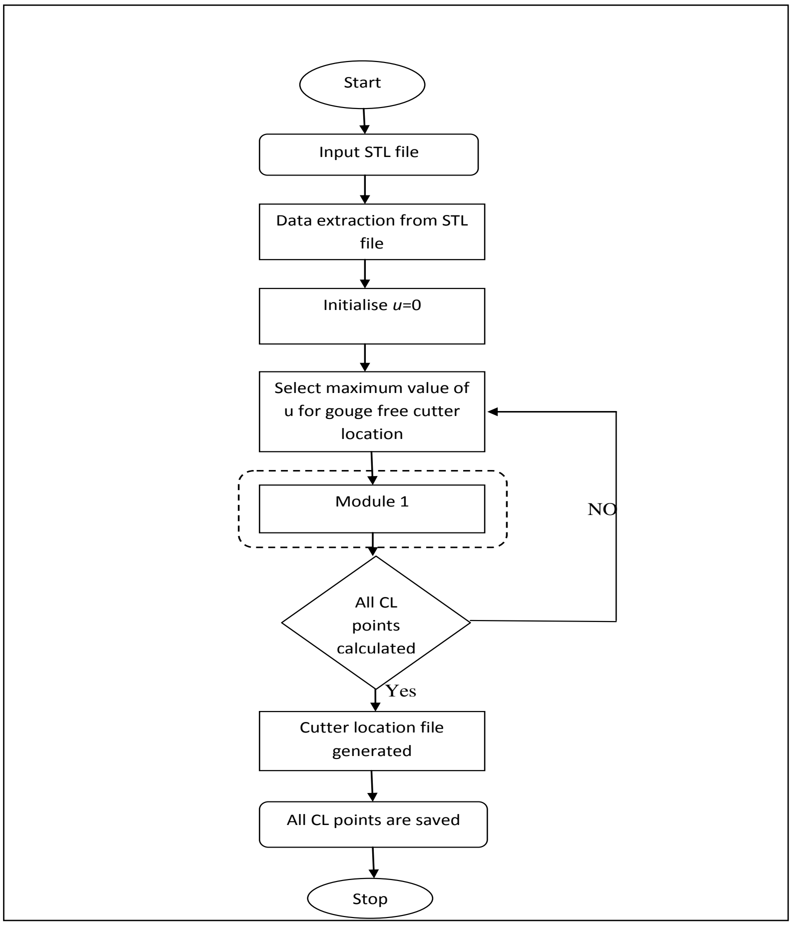

Figure 3.9: Flow chart for tool path generation algorithm

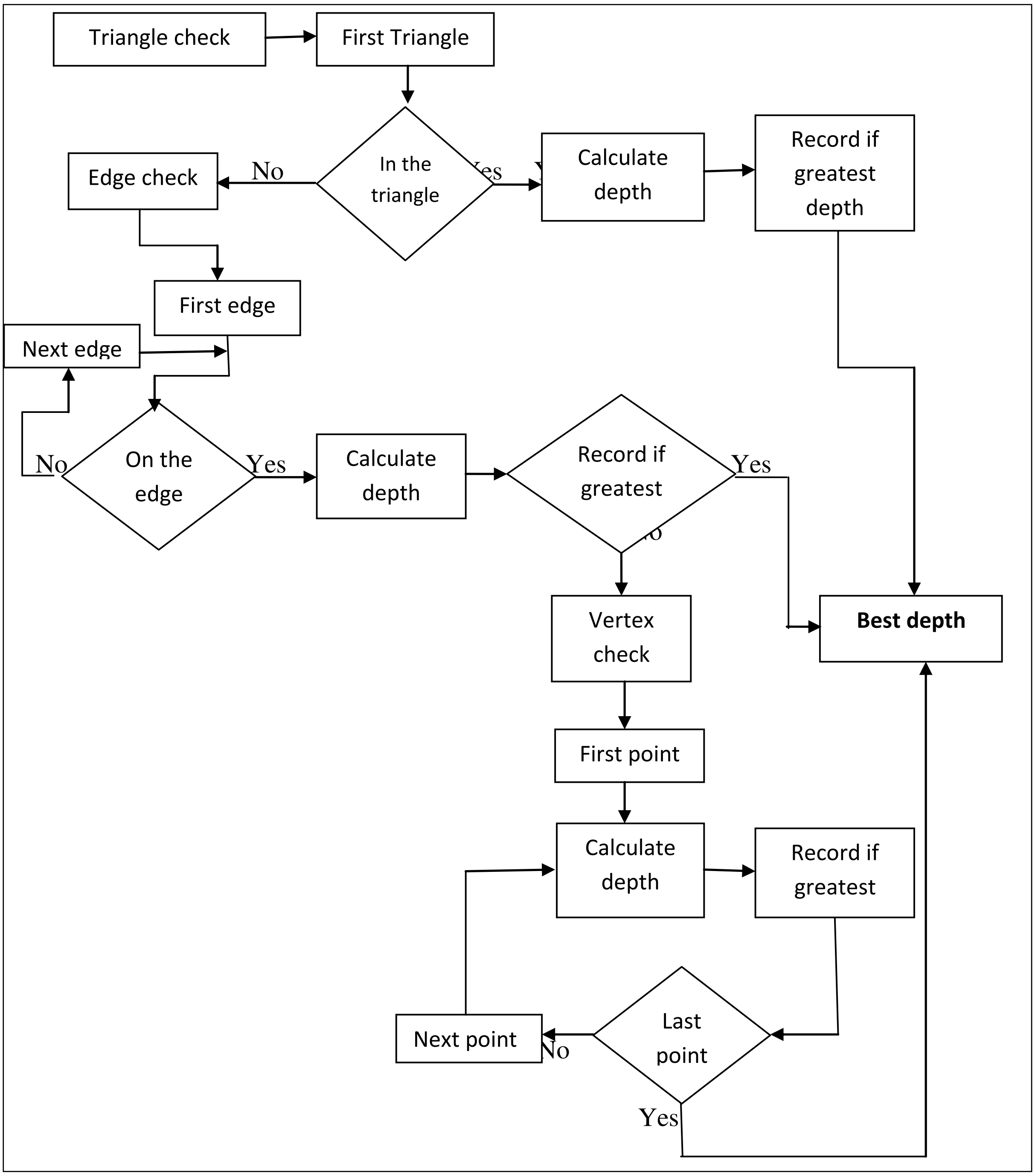

Figure 3.10: Flow chart showing working of module-1

CHAPTER 4

RESULTS AND DISCUSSIONS

The tool path planning and generation done using equations in chapter 3 has been validated and machined in actual for a sculptured surface using Bridgeport VMC 2216 3-axis vertical milling machine shown below (figure 4.1). The machine has Fanuc control and runs at 8000 rpm.

Figure 4.1: Bridgeport VMC 2216 3-axis CNC milling machine

The three axis of the milling machine are shown in figure 4.2.

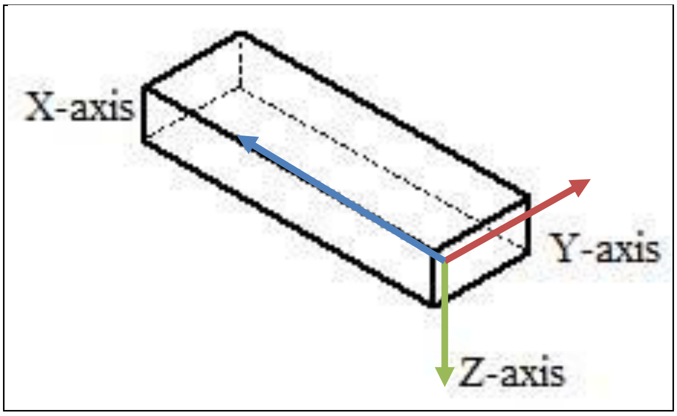

Figure 4.2: Figure showing 3-axis of vertical milling machine

4.1 TOOL USED

Three tools have been used in the experiment to validate the algorithm. After the machining process is done with each tool, surface roughness is to be checked for every surface. The parameters of the tools used are shown below.

TOOL TYPE	TOOL SHANK RADIUS (mm)	LENGTH OF TAPER (mm)	ANGLE PER SIDE
Conical end mill	10	25	14^0
Ball nose end mill	8	-	-
Cylindrical end mill	8	-	-

Table 4.1: Different tools used

The parameters used above are applied in the algorithm, but the program could run for any cutter radius. All three checks are applied on the input STL file and the next position of the tool is obtained my moving tool in a zig-zag path. Hence a tool path is generated with .tp extension. This file generated is then used in 3-axis CNC machine described above to experimentally validate the tool path.

4.2 MATERIAL USED

The material used for the experimental validation of the algorithm is Aluminium (6061).

The chemical composition of the material has been described below-

METAL NAME	PERCENTAGE RATIO OF WEIGHT(%)
Aluminium	95.15-98.56
Silicon	0.4-0.8
Iron	0.0-0.7
Copper	0.15-0.4
Manganese	0.8-1.2
Chromium	0.04-0.35
Zinc	0.0-0.25
Titanium	0.0-0.15
Other metals	0.05-0.15

Table 4.2: Aluminium (6061-T6) material composition

4.3 VALIDATION OF TOOL PATH

The tool path has been validated by machining the test part shown n figure 4.3. The dimensions of test part are $100 \times 100 \times 50$ mm. The height of the taper region is 25mm with taper angle of 14^0. The test part is machined by conical tool described in table 4.1 using the algorithm explained in chapter-3. The test part is also machined using ball nose and cylindrical end mill and then roughness of the machined surface is calculated by the use of roughness tester RTR-10. The RTR-10 uses Measurement Studio Lite (Version 3.0) software to display the results.

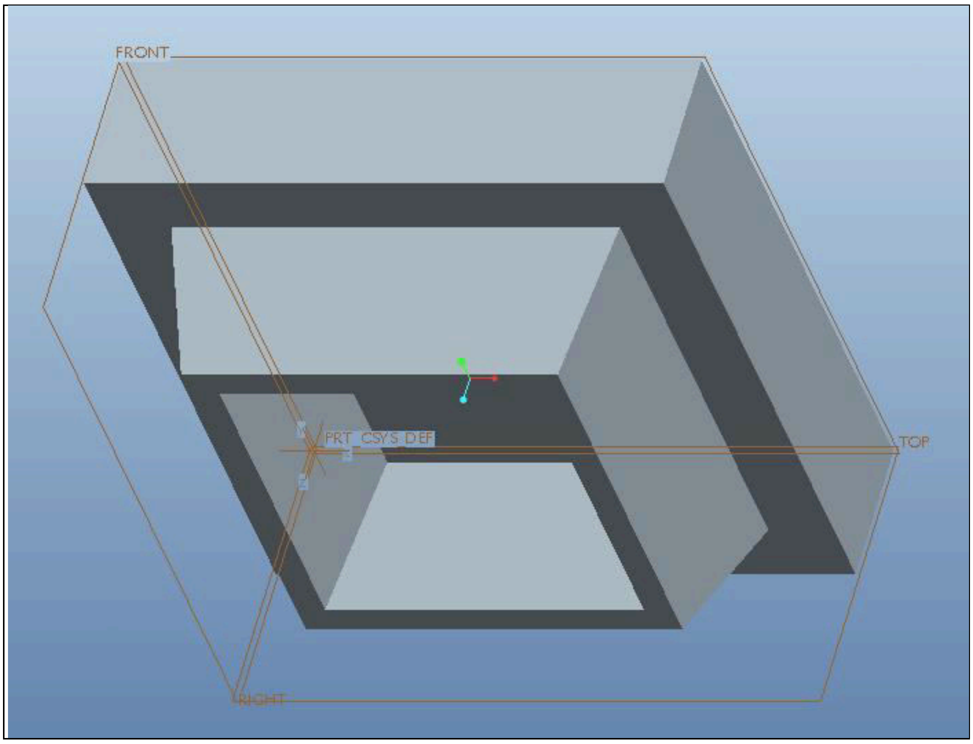

Figure 4.3: Test part

The depth of cut has been kept at 30 micron with feed of 2000 millimetre/minute and spindle speed of 3800 rpm.

4.4 EXPERIMENTAL VALIDATION

The test part is machined on the 3-axis vertical milling machine described above with three different cutters on a single aluminium plate of $350 \times 140 \times 50$ mm dimensions. The first cavity (figure 4.4) is machined using conical cutter, second cavity with ball nose end mill and third cavity with cylindrical end mill.

After machining operation is done, the surfaces are tested for surface roughness using RTR-10. The results are shown below:

4.4.1 Results obtained

Different roughness parameters are listed below.

R_a = Roughness average or arithmetic average

R_t = Maximum height of profile

R_z = Average maximum height of profile

R_c = Mean height of profile irregularities

The test part shown in figure 4.3 is now machined using three different tools and roughness then is checked using RTR-10. The results are listed below.

4.4.1.1 Using Conical end mill

Figure 4.4: Front view of cavity cut using conical end mill

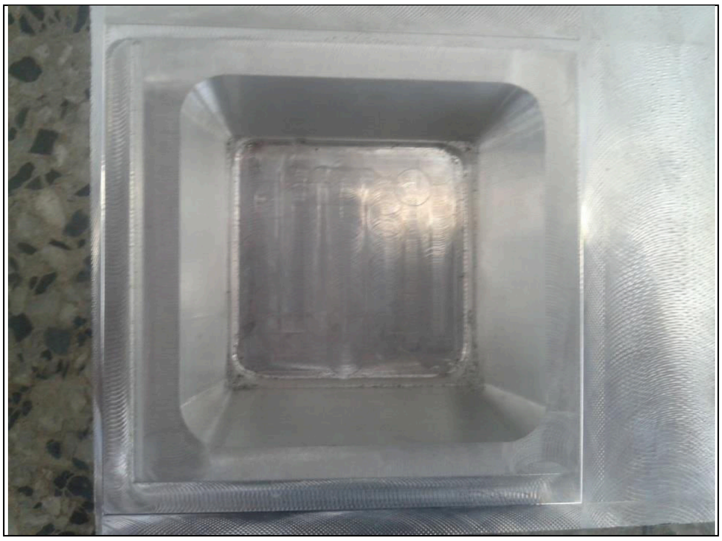

Figure 4.5: Top view of cavity cut using conical end mill

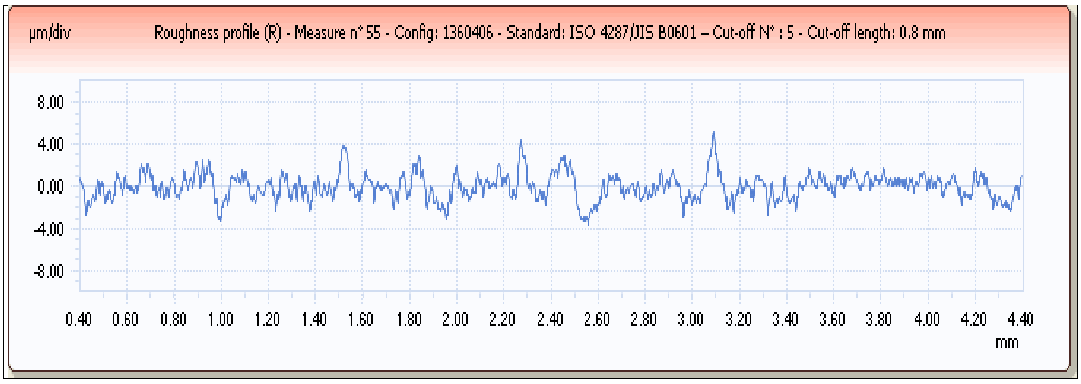

Figure 4.6: Roughness profile using conical end mill

$R_a(\mu m)$	$R_t(\mu m)$	$R_z(\mu m)$	$R_c(\mu m)$
0.91	8.92	6.62	2.23

Table 4.3: Roughness parameters obtained using conical end mill

4.4.1.2 Using ball nose end mill

Figure 4.7: Front view of cavity cut using ball nose end mill

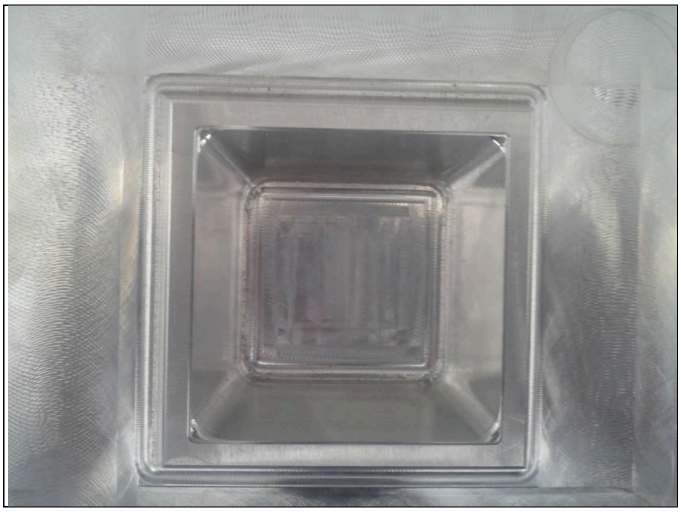

Figure 4.8: Top view of cavity cut using ball nose end mill

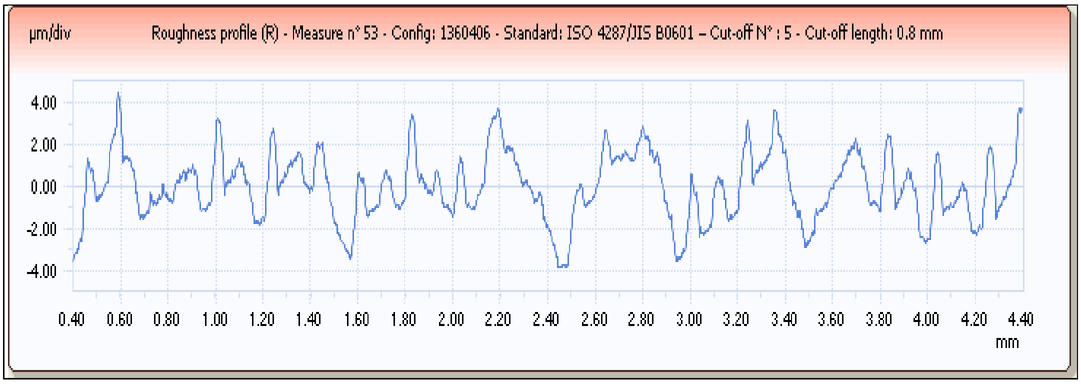

Figure 4.9: Roughness profile for ball nose end mill

$R_a(\mu m)$	$R_t(\mu m)$	$R_z(\mu m)$	$R_c(\mu m)$
1.26	8.45	7.28	3.19

Table 4.4: Roughness parameters obtained using ball nose end mill

4.4.1.3 Using cylindrical end mill

Figure 4.10: Front view of cavity cut using cylindrical end mill

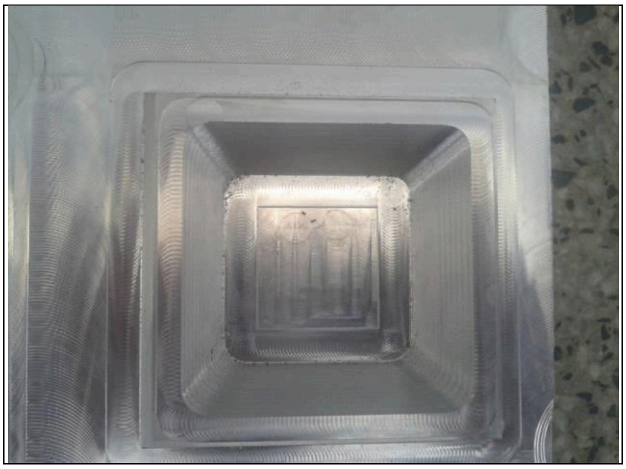

Figure 4.11: Top view of cavity cut using cylindrical end mill

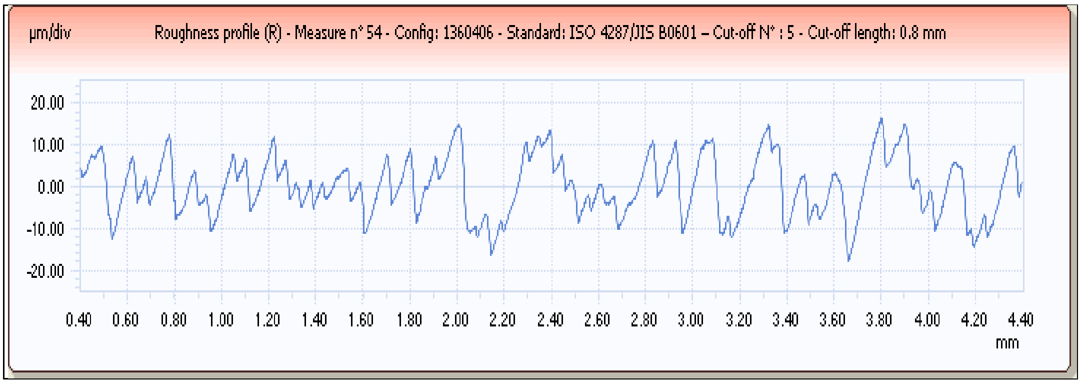

Figure 4.12: Roughness profile for cylindrical end mill

$R_a(\mu m)$	$R_t(\mu m)$	$R_z(\mu m)$	$R_c(\mu m)$
5.75	34.16	28.49	15.35

Table 4.5: Roughness parameters obtained using cylindrical end mill

On comparing the roughing profiles, conical tool profile has been found to be less fluctuating as compared to other two. Also the R_a value for conical tool is quite less as compared to other values obtained using other two tools. Hence we conclude that conical tool gives better surface finish when the inclination of the workpiece profile is either equal to or near about the angle of taper of the tool.

4.5 CONCLUSION

Based on the results shown in section 4.4, following results have been drawn.

The parametric approach which has been discussed in chapter 3 for the tool path generation has been verified. The algorithm and the tool path foot print are proved to be correct. The roughness results shown in section 4.4 tells us that the conical tool gives better surface finish compared to ball nose end mill and cylindrical end mill, for a surface having taper angle equal to taper of tool. The experiment can also be conducted for other inclinations, to check for the best surface finish obtained using conical tools.

4.6 FUTURE SCOPE

- The tool path can be modified according to various formats like IGES format, STEP format etc. The first step of generating the tool path is to convert the CAD model into STL format. During the discretization, significant error comes in the STL file. That error which comes out by converting the CAD model to STL file may be considered for future work.
- The algorithm can also be used on complicated surfaces using a conical tool because of its accessibility.
- Different tool path foot print techniques can also be tested using conical tools.
- The dynamic errors which are not being considered can also be incorporated.
- The effect of taper angle of conical tool on different materials can also be tested.

REFERENCES

[1] Chun Fong You and Chih Hsing You, "Automatic path generation method for NC rough cut machining for solid models", vol. 35, pp 161-173, year-1995

[2] Lambgrets C.A.H. *et.al.*, "Efficient automatic tool path generator for 2½ D free form pockets", vol.-29,pp.-151-157, year-1998

[3] Chih Ching Lo, "New approach to CNC path generation", vol.30,no.8, pp 649-655,year-1998

[4] Benjemaa Raouf and Schmitt Francis, "Fast global registration of 3D sampled surface using multi z-buffer technique", vol.-30, pp 649-655, year-1999.

[5] Becket E., Cuilliere J.C., Trochu F.," Generation of finite element mesh from stereolithography (STL) files", vol.-34,pp.1-17, year-2002.

[6] Park S.C. and Choi B.K., "Tool-path planning for direction-parallel area milling", vol. 32, pp 17-25, year- 2000.

[7] Chih-Ching Lo, "Two-stage cutter-path scheduling for ball-end milling of concave and wall-bounded surfaces", vol.32, pp 597-603, year-2000.

[8] Monies F, Rodonnet JM, Rudio W, "Improved position of a conical mill for machining ruled surfaces: application to turbine blades", vol.-29,pp 625-634,year-2000

[9] Monies F,Rudio W and Redonnet JM, "Comparitive study of interference caused by different positions of a conical milling cutter on a ruled surface", vol. -215(9), pp 1305-1317, year-2001

[10] Monies F, Felices JN, Rubio W, "Five-axis NC milling of ruled surfaces: optimal geometry of a conical tool", vol.-40, pp 2901-2922, year-2002

[11] Chuang Jang Chiou and Yuan Shin Lee, "Machining potential field control to tool path generation for multi axis sculpture surface machining", vol. 34, pp 357-371, year-2002

[12] Zao Chen and Peiquin Ye, "Tool path generation strategy for sculptured surface machining", vol. 127, pp 369-373, year- 2002.

[13] S.S. Makhanov *et. al.,* "On the tool-path optimization of a milling robot", vol.-43, pp 455-472, year-2002.

[14] Quing Shan Bao *et.al.*, "Study on virtual manufacturing model of a revolving milling cutter in 2-axis numerical control processing", vol.-120, pp 68-75 , year- 2002

[15]M. Szilvasi-Nagy and Gy. Matyasi, “Analysis of STL files”, vol.-38,pp 945-960, year-2003.
[16] Zezhong C. Chen *et.al.*, “Automated surface subdivision and tool path generation for 3 $\frac{1}{2}\frac{1}{2}$ -axis CNC machining of sculptured parts”, vol.-50, pp 319-331, year-2003
[17] H.S. Choy and K.W. Chan, “A corner-looping based tool path for pocket milling”, vol.-35,pp 155-166, year-2003
[18] Sang C. Park and Yun C. Chung, “Tool-path generation from measured data”, vol.-35, pp 467-475, year-2003
[19] Chiou CJ, “Accurate tool position for five-axis ruled surface machining by swept envelope approach”, vol.-36(10), pp967-74, year-2004
[20] Sang C. Park, “Sculptured surface machining using triangular mesh slicing”, vol.-36, pp 279-288, year-2004
[21] C.K. Toh, “A study of the effects of cutter path strategies and orientations in milling”, vol.-152, pp 346-356, year-2004
[22] I. Blasquez and J.F. Poiraudeau, “Undo facilities for the extended z-buffer in NC machining simulation”, vol.-53, pp 193-204, year -2004
[23] C.K. Toh, “Design, evaluation and optimization of cutter path strategies when high speed machining hardened mould and die materials”, vol.- 26, pp 517-533, year-2005
[24] V. Giri *et. al.*, “Selection of master cutter paths in sculptured surface machining by employing curvature principle”, vol.-45, pp 1202-1209, year-2005
[25] Su-Jin Kim and Min-Yang Yang, “Triangular mesh offset for generalized cutter”, vol.-37, pp 999-1014, year-2005
[26] Chen-Ming Chuang and Hong-Tzong Yau, “A new approach to z-level contour machining of triangulated surface models using fillet end mills”, vol.-37, pp 1039-1051, year-2005
[27] Zezhong C. Chen, “A Practical Approach to Generating Accurate Iso-Cusped Tool Paths for Three-Axis CNC Milling of Sculptured Surface Parts”, vol.-8, pp 29-38, year-2006
[28] Stephen P. Radzevich, “A novel method for mathematical modeling of a form-cutting-tool of the optimum design”, vol.-31, pp 2639-2654, year-2007
[29] Young-Keun Choi and A. Banerjee, “Tool path generation and tolerance analysis for free- form surfaces”, vol.47, pp 689-696, year-2007
[30] Jean Marie Langeron, Emmanuel Duc, Claire Lartiguec, Pierre Bourdet, “A new format for 5-axis tool path computation, using B-spline curves”, vol.-36, pp 1219-1229, year-2004.

[31] H. -T. Yau a, C. -M. Chuang a, Y. -S. Lee, "Numerical control machining of triangulated sculptured surfaces in a stereolithography format with a generalized cutter", vol-42(13), pp 2573-2598, year-2004
[32] Yongfu Ren, Weihang Zhu, Yuan-Shin Lee, "Material side tracing and curve refinement for pencil-cut machining of complex polyhedral models", vol.-27,pp 1015-1026, year-2005.
[33] Chaves-Jacob J, Poulachon G, Duc E, "New approach to 5-axis flank milling of free-form surfaces: computation of adapted tool shape", vol.-41(12), p918-829, year-2009
[34] LiMin Zhu, Gang Zheng, Han Ding , YouLun Xiong, "Global optimization of tool path for five-axis flank milling with a conical cutter", vol.-42,pp 903-910, year-2010.

[35] Kandarp Patel *et.al.*, "Optimal tool shape selection based on surface geometry for three-axis CNC machining", vol.-57,pp 655-670, year-2011