Web Based Automatic Tool Path Planning Strategy for Complex Sculptured Surfaces

by

Kandarp Patel

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Master of Applied Science
in
Mechanical Engineering

Waterloo, Ontario, Canada, 2010

Authors Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Over the past few years, manufacturing companies have had to deal with an increasing demand for feature-rich products at low costs. The pressures exerted on their existing manufacturing processes have lead manufacturers to investigate internet-based solutions, in order to cope with growing competition. Today, the availability of powerful and low cost 3D tools, along with web-based technologies, provides interesting opportunities to the manufacturing community, with solutions directly implementable at the core of their businesses and organizations.

The wooden sign is custom i.e. each sign is completely different from each other. Mass Customization is a paradigm that produces custom products in masses. A wooden sign is custom in nature, and each sign must be completely different from another. 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 better profit margins.

This thesis presents the automatic web-based tool path planning method for machining sculptured wooden sign on 3 axis Computer Numerical Controlling (CNC) Machines using optimal and cost-effective milling cutters. The web-based tool path planning strategy is integrate with web-based CAD system to automatically generate tool paths for the CAD model using optimal cutter within desired tolerances. The tool path planning method is divided into two parts: foot print (path along which cutter moves) and cutter positioning. The tool path foot print is developed during design stage from the CAD model based on the type of surface to be machined. The foot print varies from part to part which facilitates the mass customization of wooden sign. After designing foot print, the foot print is discretized into points and the gouge-free cutter position at each of these points is found using “Dropping Method”. The *Dropping Method* where cutter is dropped over the work piece surface, and the highest depth at which cutter can go without gouging the surface is calculated. This is repeated for all the position along the foot print. This tool path planning strategy is developed for ball nose, flat-end and radiused end milling cutter for machining wooden sign.

The tool path generated using this method is optimized for machining time, tool path generation time and final surface finish. The bucketing technique is developed to optimize tool path generation time, by isolating the triangles which has possibility of intersection at particular position. The bucketing Technique reduced the tool path computation by 75 %, and made tool path generation faster. The optimal cutter selection algorithm is developed which selects best cutter for machining the surface based on the scallop height and volume removal results. The radiused end milling cutter results in highest volume removal which results in lower machining time compared to ball nose end milling cutters, but the scallop heights is higher. However, the scallop height in the radiused end milling cutter is higher only in few regions which reduces the final surface finish. For a sign, it was found around the boundary of logo, outline of lettering, interface of border and background. Thus, in order to achieve higher surface finish and lower machining time, a separate tool path is developed using “Pencil Milling Technique” which will remove the scallops from the regions that was inaccessible by radiused end mills. This tool path with the smaller cutter will move around the boundary of logo and lettering, and clean-up all the scallops left on the surface.

The designed tool path for all the three cutters were tested on maple wood and verified against the actual Computer Aided Design model for scallop height and surface finish. The numerical testing of tool path was carried out on a Custom Simulator, ToolSim and was later confirmed by actually machining on a 3 axis CNC machine. The same sign was machined with variety of milling cutters and the best cutter was selected based on the minimum scallop and maximum volume removal. The results of the experimental verification show the method to be accurate for machining sculptured sign. The average scallop height in a machined using $1/8^{th}$ inch radiused end milling cuter and using Pencil tool path on the machined surface is found to be 0.03989 mm (1.5708 thou).

Acknowledgements

I am thankful to my supervisor, Dr. Sanjeev Bedi, whose encouragement, guidance and support from the initial to the final level enabled me to develop an understanding of the subject.

I would like to thank my family for their support and encouragement.

I would also like to gratefully acknowledge the support of some very special individuals. They helped me immensely by giving me encouragement and friendship. They mirrored back my ideas so I heard them aloud, an important for this writer to shape his thesis paper and future work. Rajnish Bassi, Tarun Kandala, Gerardo Salas Bolaos, Carlos Wang, and all other fellow researchers of PBG i.e. Professor Bedi Group. I can only say a proper thank you through my future work by helping as many researchers as possible.

Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of the project.

Dedication

This is dedicated to my family.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Need for CAM Automation

The increasing speed at which users can browse the internet has made it a profitable source for marketing and selling products to end users. The internet commerce revolution has leveled the playing field, making it possible for small and medium size manufacturers to compete for the consumers in today's fiercely competitive markets. To gain advantage companies must be agile enough to respond quickly to the demand for customized products and bring high quality, innovative, aesthetically appealing products to market at warp speeds. Optimization of both form and function with fine tuning of design process is essential in every product to gain an advantage in the market. Many companies now offer internet portals allowing users to specify product details. These details are then used to design and manufacture the products. The design and manufacturing of the product is typically done manually with the use of advanced tools. In addition, due to increasing competition in the market; there is constant pressure to reduce the actual cost of the product. The major cost is incurred at the manufacturing stage of a product, which could be minimized by automating and optimizing current strategies. Few companies have successfully attracted lot of customers by offering extra functionality in their web applications by incorporating customer specifications gathered via the web into the products. Designing custom sport shoes online is offered by Nike ID, customization of iPod is offered by Apple. Although these companies offer customization it is limited to high end products only.

1.1.1 Proliferation of Mass Customization

Mass Customization is the new paradigm that is replacing mass production, increasing by growing product variety, and opportunities for e-commerce. Mass Customization is a paradigm based on creating variety and customization through flexibility and quick responsiveness. It is the new tool in business competition for both manufacturing and service industries as it increases variety and customization with a corresponding increase in cost. Mass Customization is far more than making efficiency improvements through automated production processes, integrated supply and delivery chains and the like, though these mass production techniques are still important. It moves the focus from buying on price (because all goods and services appear to be much the same), to one of buying on satisfied needs and wants, at a competitive and affordable price [17]. Each product is unique based on customer specifications and available at affordable prices which entices customers to buy the product.

Mass Customization processes integrate the design and sales activities before the production stage. Customers become participants within the design process and help design the product to meet there need. Such a customer has been known as a prosumer as they are producer and consumer of the product. Once customers have designed their product, it is simulated and the results are shared with them over the web. This allows customers to try out the product under varying operating conditions and to redesign the product until their needs are satisfied. Once the customer confirms the design and settles the financial aspects, only then the product is built and delivered. Mass customization proactively manages product variety in the environment of rapidly evolving markets and products. Mass customization can tailor products quickly for individual customers or for niche markets with near mass production efficiency and speed. Using the same principles, mass customizers can Build-to-Order both customized products and standard products without forecasts, inventory, or purchasing delays.

Process Planning is the collection of all activities that translate a part's design specifications from engineering drawings into the manufacturing instructions required to produce it. Process planning acts as the bridge between design and manufacturing. Process planning refers to all the essential information needed to manufacture a product in such a way that all design specifications are met. With use of computerized systems process planning are simplified, optimum process plans are produced quickly and consistently; and more efficient use of manufacturing resources is achieved [7]. Still, the intricacy and interdependent nature of manufacturing processes make the effective implementation of Process Planning in industry very difficult especially those involved with cutting operations [3]. The way it is carried out also depends to a large extent, on the production situation involved. Process planning involves all manufacturing processes, such as metal cutting, sheet metal forming, welding, and assembly. Some of these processes are manual while others are done on automatic machines and rest is done on Computed Numerically Controlled (CNC) machines.

The process plan for a mass produced product is shown in Figure 1.1. For the mass production, the products manufactured are virtually identical to each other and the process plan is simple. The operator/designer generates geometric model using CAD system as the design is common for all parts, so the cost of the operator and the CAD system is distributed over the entire batch. Once the geometric model is created, the process plan comprises of manual, automated and programmed process in an order specific to the part design. The manufacturing code for programmed process must be generated. For generating the manufacturing code an operator skilled using CAM system is required. The manufacturing code is generated for the part once and the same code is used for all the other parts, as each part is virtually identical. The part is machined on a CNC machine using that manufacturing code. Same geometric model and manufacturing code generated are used for all the mass produced products. The time required to produce geometric model and manufacturing code is lower, but the variation from part to part is restricted.

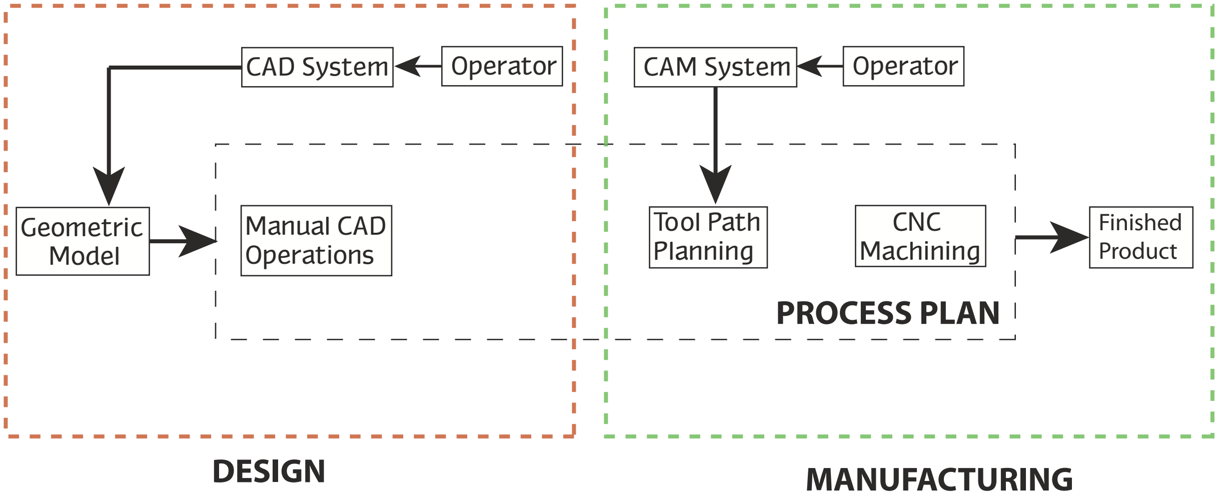


Figure 1.1: Process Planning for customized products

The traditional process plan for customized products is shown in Figure 1.2. The process plan for a product is developed after it has been designed and the part drawings are available. After studying the design, a sequence of operations to produce the part is planned. For customized products the processes, machine tools, cutting tools, measuring machines etc do not change over time. Each item of a customized product uses the same manufacturing machines in the same order. The difference between customized product and the mass product product is that each product has a different geometric model and the programmable code used to run the computer controlled machines differs from one customized part to the next. The manual and automatic operations remain the same. Currently the design description is obtained from customer and a geometric model and the programmable code for the CNC machine are produced. The manual inputs increase delivery time and cost which limits the customization to simple objects such as picture frames, key chains etc and the process to simple engraving. For generating 3D CAD models and generating code of CNC machines, a manual operator is required. The operator has to be skilled in the use of CAD and CAM software, and requires regular retraining as these software keeps evolving with time. The cost of this skilled labor and retraining is high. The complexity of the process plan depends upon the complexity and nature of the part. Although each part is completely different from other part, the process planning time is increased by the need for custom design and custom tool paths and code for CNC machines. The cost is higher compared to mass production, but it is advantageous as the flexibility of customization, appeals to customer who are willing to pay more.

The generation of geometric model and manufacturing code shown by hashed lines in Figure 1.2 are the two major operations which increase the process planning time and cost for customized product compared to mass product.

The premise in this work is that if the process plan for mass customized product can be developed in a way that reduces the time required for generation of geometric model and manufacturing code, than mass customization would increase in popularity. A web-

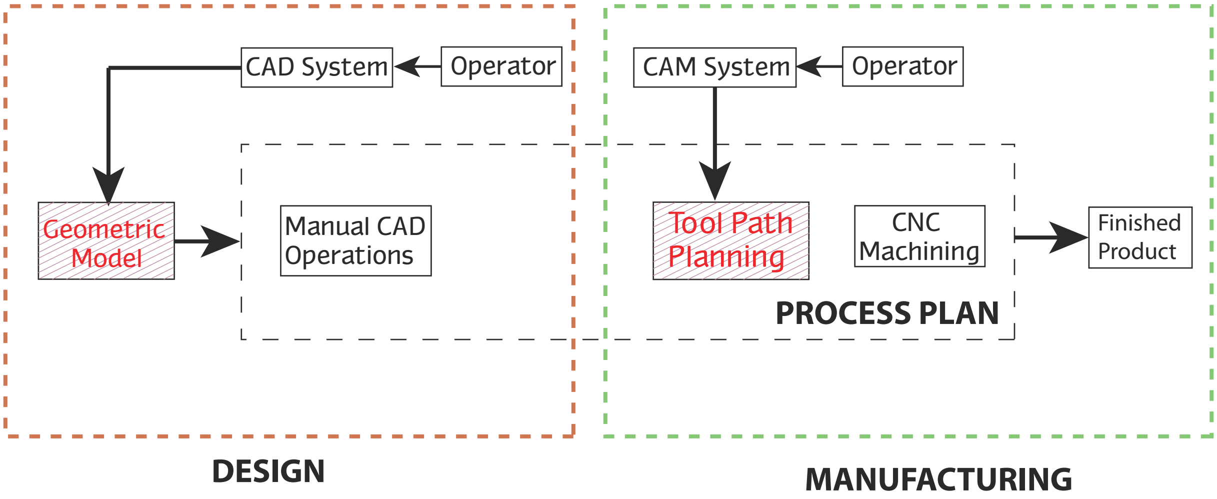


Figure 1.2: Process Planning for customized products

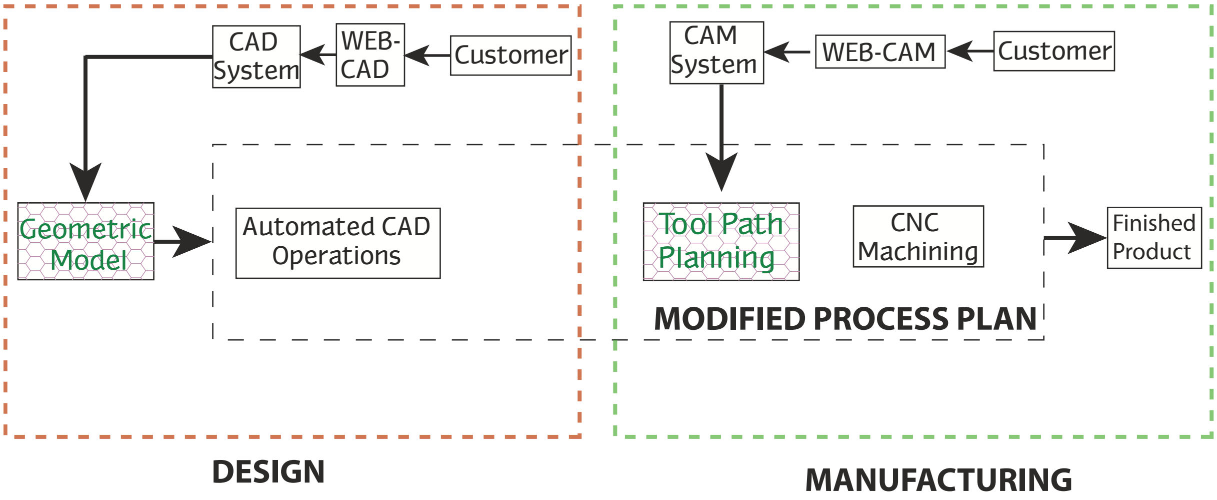


Figure 1.3: Process Planning for Mass production of customized products

based integrated CAD/CAM automation system, whose process plan is shown in Figure 1.3 offers a method to produce geometric model and programmable code for machines with little extra cost. The integrated web-based CAD/CAM automation system automates the generation of 3D geometric models and manufacturing code, reducing the process planning time for the two major time-consuming operations for customized products.

The process plan for the web-based CAD/CAM system is shown in Figure 1.3. This system replacing the skilled operator required to develop geometric model and manufacturing code using CAD/CAM system, with a web-based CAD/CAM interface custom designed to produce the geometric design of family of parts. The time required by an operator to perform manual operations during design and manufacturing stage are eliminated by developing a web-based application that takes design description from the customer in the form of parameters and values, and converts them into geometric models with the use of algorithms. This geometric model is used in a web-based tool path planning algorithm to produce programmable code for machining. The honeycomb hatched area in Figure 1.3 represents automatic generation of geometric model and manufacturing code over the internet without need of an operator using the web-based CAD/CAM system. The web interface allows customers to iterate through various designs with ease and edit the design repeatedly until they are satisfied with design without need of an operator. The web CAM system interacts with the web CAD system to automatically generate the manufacturing code for the geometric model created by web CAD system. By automating the generation of geometric model and manufacturing code required to machine products on CNC machines, the process planning time and cost can be reduced significantly. This would allow customized products to be produced in masses within justifiable time and cost.

The automated design portion of the process planning shown in Figure 1.3 was proposed by Kandala [21], while the automated web-based manufacturing portion of the process planning shown in Figure 1.3 is presented in this thesis. The focus of this work is represented by the shaded green area representing Manufacturing. The goal is to take geometry produced by a web-based CAD application and to produce tool path for the CNC operations to manufacture it.

Background Information

Complex sculptured surfaces are widely used in many engineering objects such as turbines blades, automobile components, aerospace parts and medical implants. The manufacturing of such custom intricate designs is an expensive and time consuming task. Computer aided design (CAD) and Computer Aided Manufacturing (CAM) technology has been around for decades to assist with efficient design and manufacturing of such components. With CAD/ CAM software, Computer Numeric Controlled (CNC) machines can machine intricate components with highest dimensional accuracy. CAD/CAM software simplifies the programming of CNC machines and helps improve their utilization [24].

The complete CNC machining process is divided into three stages: Design phase, Tool Path Planning stage, Verification and manufacturing stage. In the *Design* stage, a 3D Computer Aided Design (CAD) model for the design surface is created using the modeling capabilities of a CAD packages. In the *Tool Path Planning* stage, the path for the cutter is planned to produce the component from the CAD model within acceptable dimensional tolerances. In this stage, cutter location data that can be translated into G-Codes that can be understood by CNC machines is generated. This code which will guide the tool to machine the part/component. In the *Verification* stage, the generated tool path is verified and checked in a simulator for machining errors or flaws before manufacturing the component. In the *Manufacturing* stage, the actual component is loaded on the machine and the machine tool will read the G -codes generated during tool path planning stage to produce final product in its intended application.

For complex surfaces the component is further machined in three sub-stages: Rough machining, Finish machining and Manual finishing. During the initial *Roughing* cut, a large amount of material is removed to sculpt the general shape of the surface, as quickly as possible without bringing the tool in contact with the desired surface. A large amount of material is removed in a short span of time. After roughing, *Finishing* cuts are used to remove the desired surface. Typically a ball nose end mill is used to generate the desired surface. These operations results in large number of scallops. Grinding and polishing, called *Manual Finishing*, is used to remove these scallops. For some applications Semi Roughing and Semi Finishing stages are used to achieve the desired accuracy.

Conventionally, complex surfaces are machined using ball nose end milling cutter on 3-axis milling machines. CNC milling machines direct the tool via a computerized controller. The computer controller acts like an interpreter, reading program statements, executing them and directing the specified actions to take place. A specific part is machined by moving the cutter along a pre-computed path in space called tool path. The path is part of a NC program that contains several other elements as well: the feed rate at which to move the cutter through the work piece, rotational speed for the spindle, instruction for tool change etc. Cross feed and feed forward distance are also elements that influence a NC tool path, although they are not part of a NC program [11].

1.2 Research Scope

The focus of the current work is to offer an innovative solutions for simplifying the generation of programmable code for manufacturing process to produce mass-customized product. The solution for simplification is part of a web-based design and manufacturing systems which assists in automation of CAD and tool path planning. The current programmable codes for machines only stores data related to movement of its axis and stores no design information. Although the point where a programmed code is generated a CAD model with complete geometry and topology is used. This loss of information regarding topology and dimensional accuracy of the surface, ultimately results in various types of machining problems, such as, gouging, over-cutting, higher scallops, and poor surface finish. The web-based system integrates CAD and CAM information, and does the design and tool path planning in an integrated environment.

A method for designing objects over the internet was proposed by Kandala [21]. He argued that the customer know the details of the product they want and are thus the key candidates to provide this information. He went further to propose that instead of using customers to provide all product data they should be asked for only relevant details. Based on this he developed the idea of web-based CAD system targeted to a family of components. Within the predefined scope of the geometric family infinite variations can be created based on customer requirement.

The system developed by Kandala as shown in Figure 1.4 has a predefined algorithm written in the back-end of the web-based application designated by the CAD box in the Figure 1.3, with a user interface for data exchange with the customer. Based on the user inputs, the algorithm would generate 3D CAD model and communicate visual models to the user/customer over the web browser. The interface allow user to redesign the part till design is to their liking accepted. It is proposed to use the system developed by Kandala [21] to implement and test the idea of automatic generation of programmable code for machines (designated as CAM in Figure 1.4). The CAM algorithm is initiated by CAD macros, which will analyze the 3D CAD model generated by the user and create an optimal tool path automatically. Based on the work piece to machined, the CAM algorithm will select the type of cutter, number of roughing and finishing passes required, feed rate, depth of cut etc. The tool path is directly fed in the CNC machines to manufacture a component. The detailed system architecture of the web-based automatic design and manufacturing solution is shown in Figure 1.4. The scope of all customizable parts is large. In the system

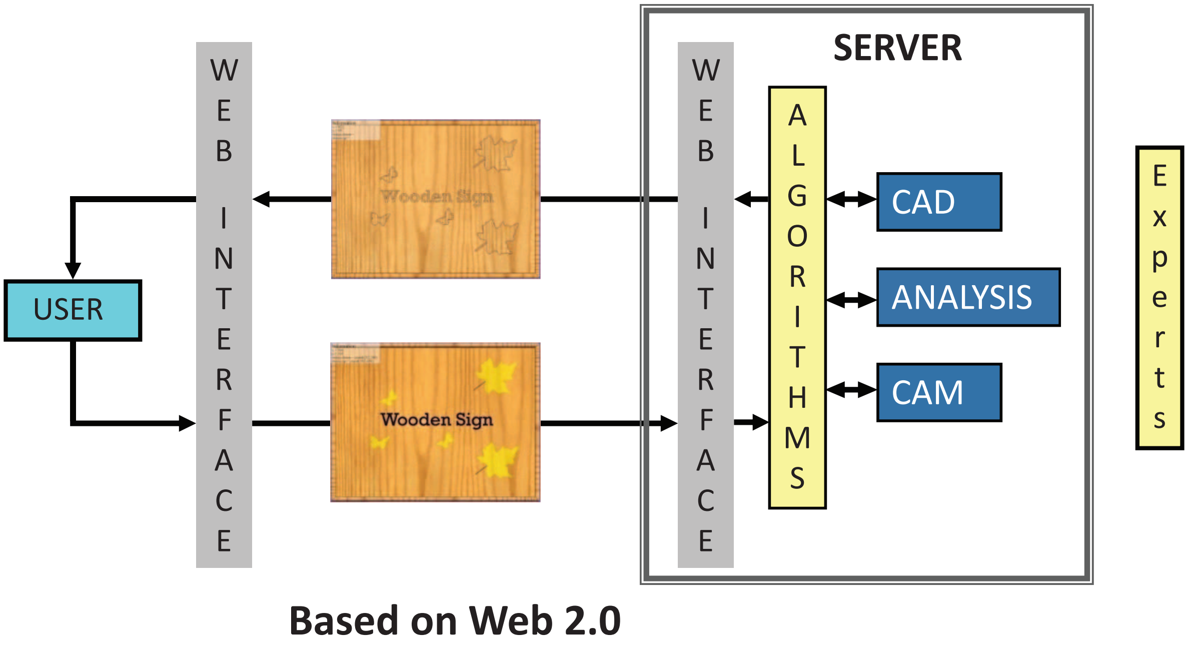


Figure 1.4: Technology Model for Mass customization of Wooden Sign

designed by Kandala [21], the geometric family is limited to a wooden sign comprising of a base shape, border, logo, background and text. It is proposed to use Kandala's system for automatic generation of tool path which can be used on a CNC router. The automatic tool path planning system will analyze the CAD model generated from CAD system and will generate tool path which will machine wooden sign with specified surface finish and minimal machining time. This system connects users and the experts as shown in Figure 1.4 and makes it possible to take advantage of new machining concepts developed by experts.

The detailed structure of the CAM automation system is shown in Figure 1.5, which would generate manufacturing code over the internet. The integration of CAD system with CAM system and the flow of information are shown by dashed boxes in Figure 1.5. Kan-

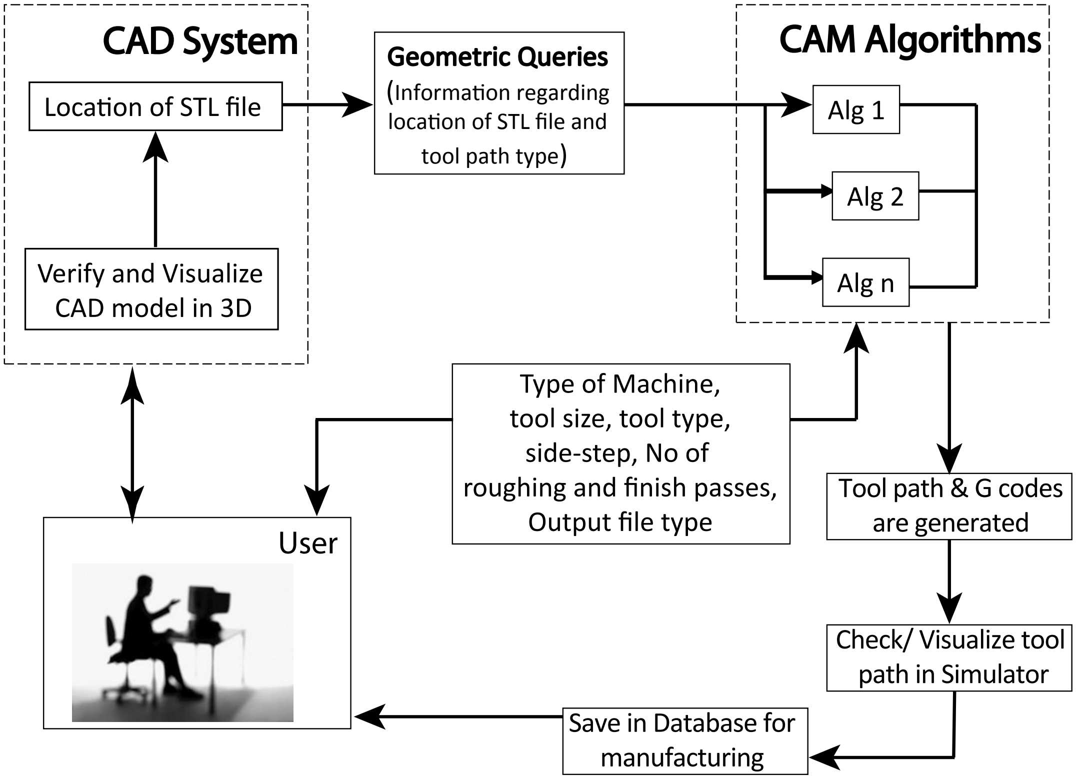

CAD System
Location of STL file
Verify and Visualize CAD model in 3D
Geometric Queries (Information regarding location of STL file and tool path type)
CAM Algorithms
Alg 1
Alg 2
Alg n
Type of Machine, tool size, tool type, side-step, No of roughing and finish passes, Output file type
User
Tool path & G codes are generated
Check/ Visualize tool path in Simulator
Save in Database for manufacturing

Figure 1.5: CAM system for mass customization of Wooden Sign

dala's system integrates web-based CAD/CAM system for the mass customized product by converting a complex design of a sign into different features. The user will input design parameter of those features in the system and these parameters would be converted to a CAD model. Once the CAD model is complete it will initiate the CAM application, and the CAD system will convert the CAD model into STL file (a file that stores information of CAD model in form of triangles), which is uploaded onto the CAM system for generating tool path for machining. The CAD system will pass geometric queries to the CAM system. The geometric query contains information regarding the location of STL file, shape and size geometric model, and the location in the database where tool path should be saved. The CAM system will get information from the user for generating tool path specific to the part designed by the user. The information regarding the type of CNC machine, size and shape of cutter, side-step, number of roughing and finishing passes etc is taken from the user. The CAM algorithm will process this information and generate tool path, This tool path is checked in a simulator and later saved in the database for machining. Such a web-based strategy will reduce the lead time, design and manufacturing costs.

1.3 Research Challenges

The goal of this research is to create a web-based CAM system which automatically generates tool path from the parametric 3D CAD models over the internet with little or no human input. This system will be integrated with Kandala's web-based CAD system and a web-based 3D visualization for developing a fully functional web based sign design and manufacturing system. This system integrates work flow with design and manufacturing, allowing implementation of best practices and integrating them into a sequenced software algorithms. The CAD algorithms invoke CAM macros and supply them with the collated design data created by the customer. The CAM algorithm processes the CAD model and generates a tool path for manufacturing that product on the CNC Machine. In addition, some extra information regarding the machine and tooling is collected through web interface to generate tool path compatible to the customer's machine. All this is done with a predefined sequence representing the work flow in design and manufacturing of signs. User defined parametric features with CAM automation offers an innovative way for non CAD/CAM users to design their 3D models and manufacture them without prior training.

The major objectives of this research are

1. Develop a computer algorithm for positioning various end milling cutter on the designed surface for the purpose of generating a set of cutter location points for a variety of tool shapes for machining. The algorithm is designed for 3 most commonly used cutters, namely, ball nose, flat-end and radiused end milling cutters.

2. Develop scallop height estimation algorithm and estimate the scallops leftover when a part surface is machined by a cutter. The algorithm will enable comparison of the three milling cutters and identify one which results in best quality of surface after machining.

3. Machine several test surfaces to test the CAM algorithms and their ability to select the optimal set of cutters.

Several research projects have been undertaken for developing tool path planning for end milling cutters and these are critiqued in the next chapter.

1.4 Thesis Layout

Chapter 2 presents *Ball-Drop method* for 3 axis CNC milling machine which is generalization of “Drop the Ball” Method developed by Manos [14]. The problems of this method for machining a custom wooden sign are discussed and solution to those problems is proposed. Chapter 3 presents previous research conducted on tool path planning on various end milling cutter and selecting optimal cutter based on geometry of the surface. Moreover, it presents survey of work related to gouge detection and correction algorithms, and tool path optimization methodologies.

Chapter 4 presents an innovative *Dropping method* for cutter location use to generate gouge-free tool path for machining the designed surface followed by a discussion and implementation. In addition, it includes tool path optimization techniques and publishing the tool path over the web

Chapter 5 presents several tool path optimization techniques for *Drop the Doughnut* technique on sign and simulation results obtained by machining a sign with the methodology discussed in Chapter 3.

Chapter 6 compares the tool path generated by several milling cutters explained in Chapter 3 and select optimal milling cutter based on the geometry of component to be machined. It includes scallop height detection algorithm and Volume removal algorithm to compare results

Chapter 7 discusses the conclusion and future research opportunities in this field.

Chapter 2

Ball-Drop Method for manufacturing custom sign

A wooden sign is one of the application that by its very nature is customer specific. Each sign has a unique name and/or logo on it. If it is coupled with the option to change sign shape, border and background the variations are infinite. The interface for sign manufacturing application with these selection of customizable features is shown in Figure 2.1. The selectable features in this application result in a complex 3D model of the sculptured surface that can only be cut using a CNC machine. A typical way to produce CNC code for machining the sign is to select geometric entities like a letter and to produce code to machine it. Such a plan cannot work in an automated system as it requires manual selection. In this work the availability of CAD information is used to assist tool path generation. This is accomplished by breaking the tool path into two parts: defining a tool path foot print; and positioning a tool on the surface in a gouge-free mechanism.

Several commercial CAM packages like Mastercam are available to generate tool paths with desired accuracy. The input data to the Mastercam is presented as a cloud of points and it fits spline curves to the acquired boundary points. The selected foot print is automatically closed after the first point, the last point and the contour direction is specified. The path along which cutter moves is then processed by the Mastercam and the CNC tool path program is generated. Most of this commercial CAM packages have fixed path (zig-zag, spiral etc) along which cutter is moved to generate tool path independent of part being machined, which limits their usage to generate tool path for the wooden sign application. For a mass customized sculptured wooden sign, the surface geometry will vary from part to part. Using a commercial CAM packages with a fixed tool path movement it would leave more material on the work piece surface which will result in poor finish.

Redonnet *et al.* [8] developed a cutter positioning strategy which places a cutter in proper orientation to machine a sculptured surface effectively. The cutter orientation is based on the curvature information in the cutter contact point. This method do not take into account the surface deviation in the neighbourhood for calculating cutter contact point e.g. while machining a detailed surface with a large cutter, some details on the surface is cut away unintentionally. Moreover, it is iterated many times in order to get best cutter position. These problem limits the use of cutter positioning strategy by Redonnet *et al.* [8] for manufacturing sculptured wooden sign.

Another CAM system named WebNC was developed for feature based product modeling and intelligent process planning to manufacture prismatic parts on 3 axis CNC machines [20]. WebNC has Feature Based Modeler (FBMod) and Intelligent Process Planning (webCAPP) module which automatically generates efficient, error free CNC programmable code from the feature based CAD model. The CNC code is post processed to FANUC controller and neutral data formats to enable running it on any industrial CNC machine.

WebNC uses Communication module allowing “Anywhere Anytime” connectivity between globally distributed users clients) and CNC machines. The WebNC deviated from the norm and developed an internet based manufacturing system to cut components on CNC machines. A personal computer with an internet connection is necessary to use WebNC system. Even though WebNC can machine part effectively with higher productivity, but part feature families used for designing is limited to free-form surfaces such as holes, pockets, slots, steps, array patterns derived from actual industrial parts. This limits there usage for generating tool path for sculptured wooden signs.

Manos [14] developed “Drop the Ball” method which could be used to machine a sculptured surface on 3 axis CNC lathes. This method can machine sculptures surface by dropping the ball nose cutter shaped like a sphere. The tool is dropped at each position, independent of the type of surface being machined which makes it efficient for machining complex surfaces. The tool is then moved over the fixed helical foot print to generate gouge-free tool path. The *Drop the Ball* method was developed and tested for machining complex sculptured surface on CNC lathes. This *Drop the Ball* method can machine sculptured surfaces efficiently, but this method is limited for the use of ball nose end milling cutter with a fixed path along which cutter is moved.

The tool path planning method developed by Manos *et al.* [12] was selected for machining sculptured wooden sign because it can machine sculptures surfaces and complex features efficiently. Moreover, the “Drop the Ball” Method can position the cutter gouge-free over the work piece surface to machine a surface.

Although different tool path can be created, a zig-zag path was used to produce the entire sign with all its features incorporated in it. This simplified the tool path planning to just developing a gouge-free tool positioning method that can be used to position the tool at a number of points and to move the tool between these points to machine the sign. For generation of automatic tool path in wooden sign application, a “Ball-Drop” Method for 3 axis CNC milling machine which is generalization of “Drop the ball” Method developed by Manos [14] for Single controlled axis lathe machines was chosen. The *Ball-Drop* method was limited to generating tool path for lathe machines with single controlled axis and only using ball nose end mills. The *Ball-Drop* method will generate tool path as the tool moves along a helical path on a lathe. For machining a component on a 3 axis milling machine, the concept of determining the cutter location was used from *Ball Drop* method, but the tool was moved along a zig-zag path instead of helical movement.

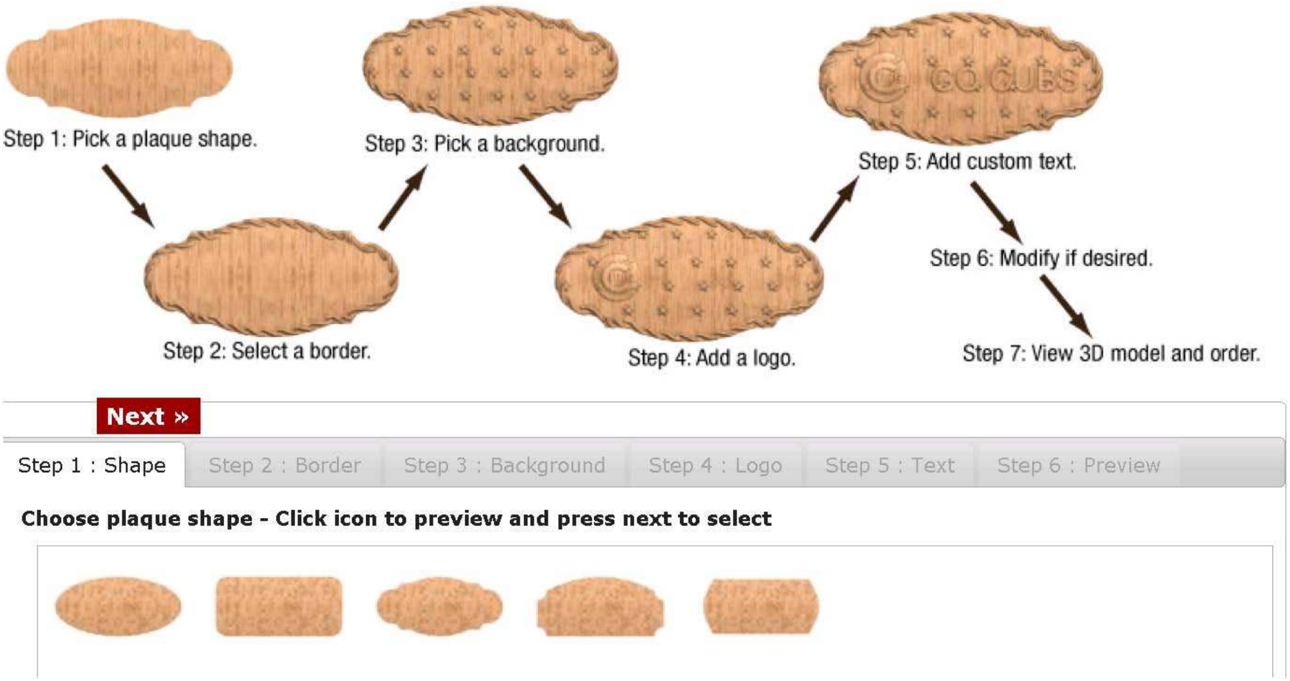
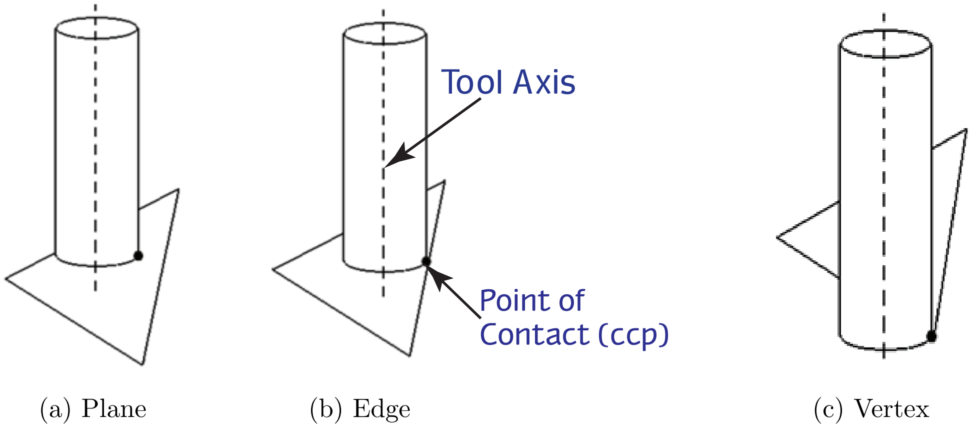

Step 1: Pick a plaque shape.
Step 2: Select a border.
Step 3: Pick a background.
Step 4: Add a logo.
Step 5: Add custom text.
Step 6: Modify if desired.
Step 7: View 3D model and order.
Next »
Step 1 : Shape
Step 2 : Border
Step 3 : Background
Step 4 : Logo
Step 5 : Text
Step 6 : Preview
Choose plaque shape - Click icon to preview and press next to select

Figure 2.1: Web Interface for Wooden Sculptured Signs

Here, the ball nose end mill was considered as a sphere above the surface that is dropped along the tool axis at a given location. The first contact point between the ball and the surface is desired as this represents a cutter location in machining the design surface. Based on intersection of the ball with the triangle a check is performed to calculate the exact cutter location.

In the *Ball-Drop* Method the cutter is restricted to follow a zig-zag path around the work piece which establishes a fixed tool path that covers the entire area. For machining any surface, the zig-zag path is discretized into closely spaced points along this tool path. The required depth at each of these points is determined through three possible scenarios for the point of contact between cutter and surface: 1) contact on planar surface, 2) contact on edge, and 3) contact on vertex. The point of contact will be dependent of the topography of the surface. For each of the three cases the highest tool position is chosen. A brief description of the checks is given in Figure 2.2. As mentioned by Manos *et al.* [12] the three checks are variations on standard closest-point computations. The Interference (i) is easier to detect because the centre of the ball always lies along the surface normal at the cutter contact point (ccp) at a distance equal to the radius of the sphere as seen in Figure 2.3 [15]. The distance d is calculated using geometry of sphere and the equation of



(a) Plane (b) Edge (c) Vertex

Figure 2.2: Cases for first contact point with the surface

the sphere, Equation 2.1, is used to detect interference. If d is negative, the coordinate is inside the sphere; if d is positive, the coordinate is outside; and if d is zero, the coordinate is on the surface of the sphere. Ball nose will never gouge the surface, if the radius of the ball is smaller than the minimum radius of curvature of the surface [15].

$$d = \sqrt{(x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2} - R_{ball} \tag{2.1}$$

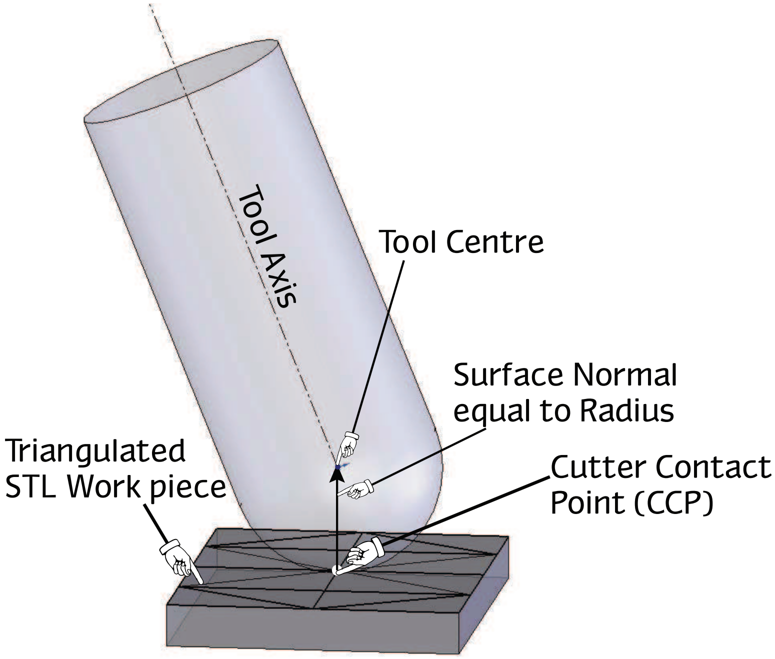


Figure 2.3: Geometry of Ball Nose Tool Position

Based on the interference of the ball with the work piece surface, the final heights of the cutter is calculated for each of these checks. Once the final depth is detected, and to generate the tool path, the tool visits these sequenced points at the calculated heights. The points visited by the tool are passed into the CNC code and stored in a file. This file is used for controlling depth of cut while machining the surface

In “Ball-Drop” Method, to achieve desired surface finish for sculptured surfaces either smaller ball nose end milling cutters are used or the step-over distance (side-step) is reduced. These options would machine a surface with desired accuracy, but would increase the machining time and ultimately manufacturing costs.

A fully sculptured wooden sign was machined using *Ball Drop* Method and is shown in Figure 2.4. The sign has a rectangular shape, circular border, plain background, a logo and text. This sign was machined using $1/8^{th}$ inch finishing Ball nose end mill with zig-zag tool path foot print and step over distance of $1/64^{th}$ of an inch. Even though, this method can machine sculptured surfaces with desired accuracy, the machining time and the computational time for generating tool path is very low. This makes it difficult to implement “Ball-Drop” method for machining sculptured surfaces like wooden sign. The problems observed in the machined sign are discussed in detail in Section 2.1 of this Chapter. The problems are critically analyzed and their solutions are developed.

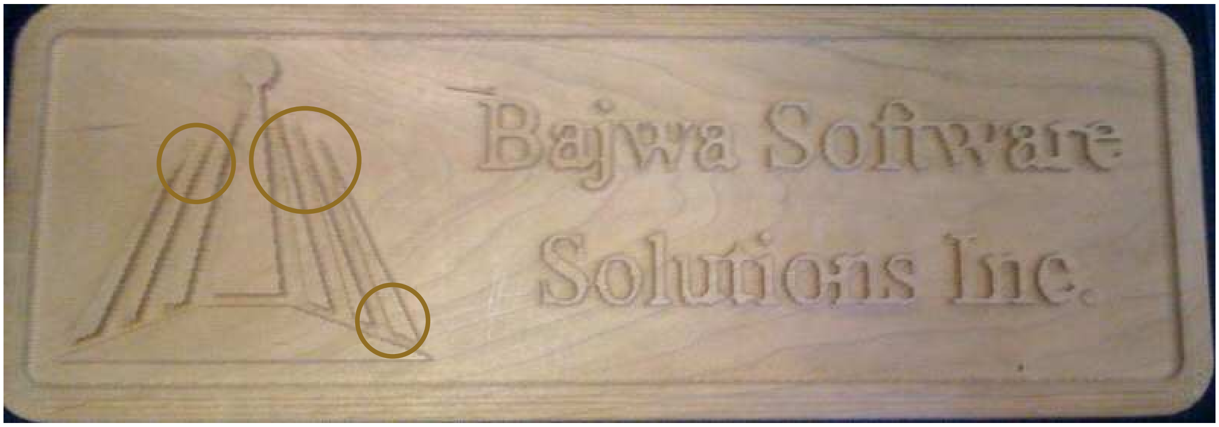


Figure 2.4: Machining results of wooden sign using Ball-Drop method

2.1 Problems with “Ball-Drop” Method

The three major problems observed while machining a sign using *Ball-Drop* Method were high machining time, huge scallop heights and burrs on the surface. These problems are discussed in detail in the following sections.

2.1.1 Machining Time

Time to Machine Sign

For mass customization of a wooden sign using *Ball-Drop* method, the machining time was very high. The time required to machine a fully sculptured sign with a ball nose end mill and zig-zag tool path was very high. For a 17” x 6” x 1” wooden sign, with $1/2^{th}$ inch and $1/8^{th}$ inch ball nose cutter for roughing and finishing operations, and with step over distance of $1/64^{th}$ of an inch took 150 minutes for machining. The major reason for high machining time is that, the tool moves over the work piece in the same foot print independent of the complexity of the surface. Moreover, the *Ball-Drop* method is limited to machine a sign with a ball nose end mills which removes lower volume of material in one pass and requires larger number of passes to machine a surface to acceptable tolerances. Unless the machining time can be reduced this paradigm will be of limited use.

Time to Compute Tool Path

In this method, the computational time required to generate tool path varies directly with the number of points in the tool path. There are many intersection calculation and checks that need to be performed in order to calculate gouge-free cutter location using *Ball-Drop* method. Increasing number of data points that represent the work piece surface increases the computation time for calculating cutter location. Moreover, in the *Ball-Drop* method for each tool position all the triangle of the work piece are checked, but the cutter will intersect only a small percentage of these triangles.

2.1.2 Surface Burrs

The final surface finish produced using the *Ball-Drop* method is very poor. Machining burrs are observed on the work piece edges after machining a sign on a CNC milling machine. The variations in cut surface quality and burr height is related to the type of cutter and direction of cutter movement. During the machining of the sign, severe burrs were observed in the background, around text and in the logo. The example of burrs in a machined sign is shown in Figure 2.5 and Figure 2.1. Burrs increases the hand sanding required to finish the sign.

Another problem experienced while machining wooden sign with a zig-zag tool path foot print was chipping of the wood where the thickness of the feature is small. The major reason for chipping is that the cutter is moving along a zig-zag path with high spindle speeds. This results in larger forces on overhang and unsupported regions leading to tearing of the wood which results in a bad finish and undesirable loops.

The chipping off in a machined sign was observed around the boundary of the logo and on the text. The chipping off in text is severe and huge amount of extra material is removed near the boundary of the letters. A comparison of actual CAD model and actual machined sign with chipping off around the lettering is shown in Figure 2.5. Similarly, the chipping off was observed at the top part of the logo where the thickness is very less. Machining such

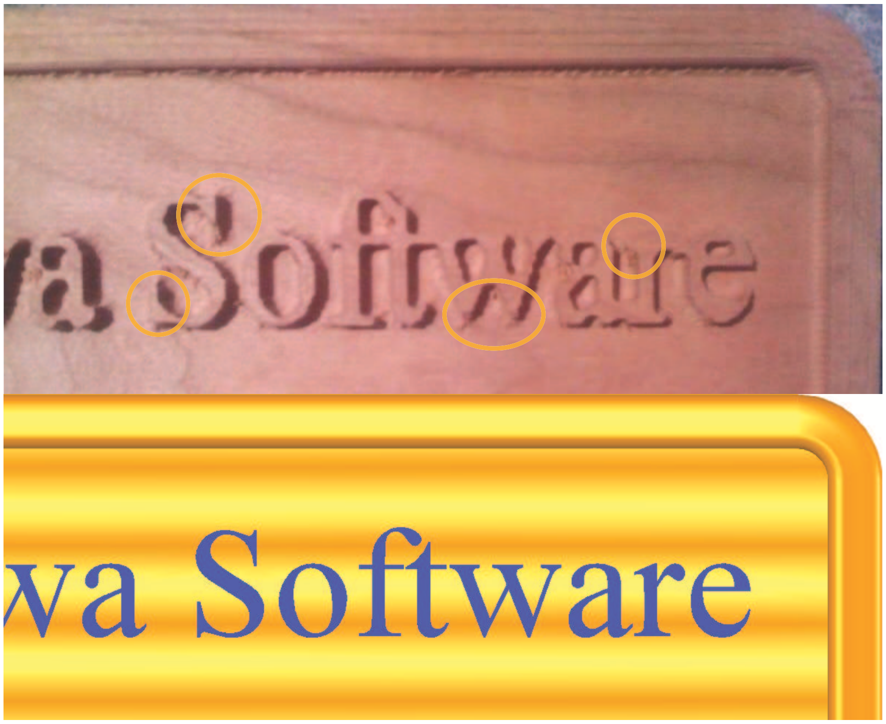


Figure 2.5: Comparison of actual machined sign with 3D CAD model

features of the sign separately with proper tool path foot print could eliminate chipping off and produce a sign with better surface finish and aesthetics.

2.1.3 Scallop heights

The surface of a wooden sign has a complex geometry. Machining this sign with a ball nose cutter results in high scallops throughout the surface and across the boundary of some features. There is huge amount of material that is left-over on the work piece between two consecutive tool passes. The problem is severe around the boundary of logo, lettering and on the border. The high scallops regions, around the logo and lettering are shown in Figure 2.6 and Figure 2.7a.

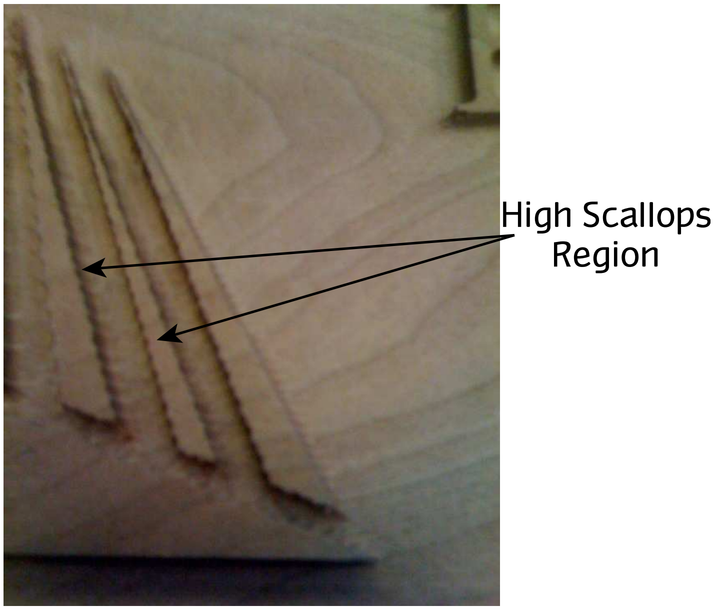
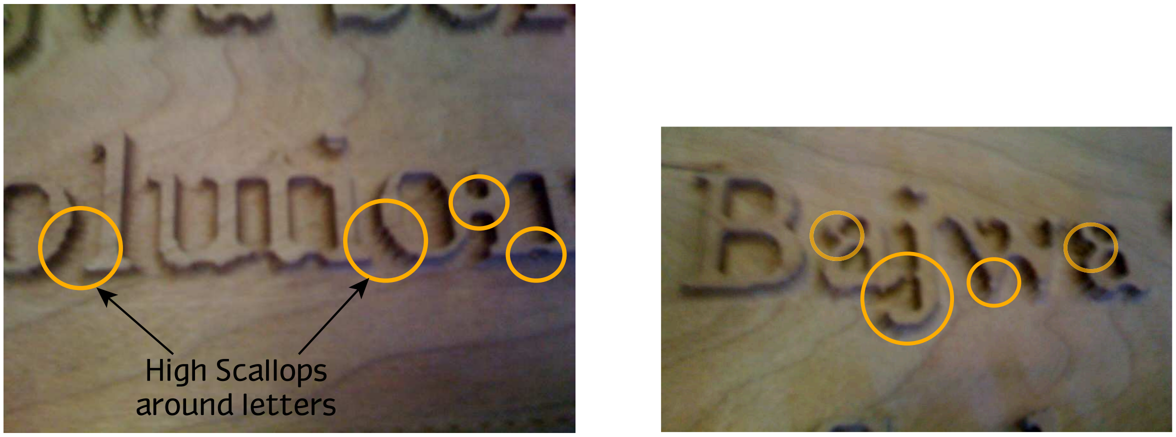


Figure 2.6: High scallop heights on the boundary of logo

Maximal scallops were observed at the locations where the cutter movement was nearly perpendicular to the surface of feature being machined. Moreover, zig-zag path leads to the formation of steps instead of smooth inclined surface. The step formation and other errors observed in a machined sign are shown in Figure 2.6. These steps represent material left-over after machining a sign. With proper selection of cutter and tool path foot print such problems can be avoided.

Another problem with machining of lettering is the spacing between them. As letters are placed, they begin to interfere with each other. The spacing between the letters is small and does not allow ball nose cutter to plunge between them which results in interconnected lettering. The example of such letters is shown in Figure 2.7b.

In summary, the *Ball-Drop* method results in higher machining time, high scallop heights and surface burrs which ultimately reduces the final surface finish and aesthetics. The machining time required to machine a sign can be improved by employing radiused



(a) Scallop regions

(b) Font Interference

Figure 2.7: High scallop regions and font interference around machined letters

end mills and flat-end milling cutters. The radiused end mills and flat-end mills can remove higher volume of material in one pass compared to ball nose end mills. Thus, the radiused end mills and flat-end mills would require lower passes to machine a part. This would allow the part to be machined with larger side-steps to achieve the same surface tolerance that could be achieved using ball nose end mills with small side steps. The computation time for the *Ball-Drop* method can be reduced by using a Bucketing technique which would divide the whole surface into small buckets, and would only check the triangles which has possibility of intersection with the cutter. This method was implemented and tested for this work and is discussed in Section 4.1.

The surface burrs generated while machining a sign with a zig-zag tool path foot print can be reduced by developing a custom tool path foot print based on the shape of the feature. The custom foot print will move the end milling cutter along a specific path based on the shape of feature, in order to machine a feature with best surface finish. The method for developing custom foot print was implemented and discussed in the Section 4.5.

The high scallops observed while machining a font using *Ball-Drop* method can be avoided with a profile tool path foot print. The profile tool path foot print will move around the boundary of the text and clean-up all the scallops around the boundary of lettering. This profile tool path foot print can be used around the logo to remove high scallops. Moreover, by using profile tool path foot print the steps formed around the logo and lettering can be removed. The method for developing profile tool path foot print was discussed in the Section 4.5. These are some deficiencies in the *Ball-Drop* method which must be resolved before it can be used effectively in industry.

A method is developed which will break the tool path into two parts: tool path foot print and gouge-free cutter positioning on the surface. By separating tool path planning into foot print and cutter positioning, a different foot print can be the tool path can be produced for different type of cutter for the same foot print independent of the

Thus, by adopting advantages of all the above mention methods, a full fledged CAM system can be developed which can machine the sculptured surface with better surface finish and minimum machining time. The "Dropping Method" is developed by uniting the "Drop the Ball" method developed by Manos [14] with radiused end milling cutter, moving it along custom foot print based on the work piece surface and making it available "Anywhere Anytime" over the internet. The "Dropping Method" drops the cutter along the tool axis and moved over the foot print for generating tool path. The web-based interface will get information from the customer and generate programmable code compatible to the user's CNC machine for machining a geometric model with best surface finish within minimum time

Thus, there is a need to generate a tool path strategy for radiused and flat-end milling cutters with a custom tool path foot print which would machine different features of a sign with best surface finish. To address these issues, the "Dropping method" was developed. The *Dropping Method* is developed for machining it with a Ball nose, flat-end and radiused end mills. This method will generate a profile tool path for the machining fonts and logos, and will also select optimal cutter required to machine a sign with best surface finish. The *Dropping Method* can machine wooden sculptured sign using optimal cutters with best surface finish and minimum machining time on 3 axis milling machines. The goal of this thesis is to generate a *Dropping Method* which would machine a wooden sign with best surface finish and minimum machining time, so that a custom sign could be mass produced.

Chapter 3

Tool Path Planning

A computer algorithm has been developed and tested for locating a ball nose, flat and radiused end mill onto a triangulated surface, for manufacturing on 3 axis CNC milling machines. The "*Dropping Method*" is presented in this Chapter which drops the cutter along the tool axis over the surface of the work piece. This method is generic and flexible, that it could be easily adapted to any type of tool and machine configuration. The traditional approach to tool path planning is to describe the tool trajectory in terms of the cutter contact point. The cutter contact point is either directly or indirectly specified. As the tool is positioned to touch these points, the cutter location is used in tool path planning. These method do not have an inbound gouge avoidance strategy and use of such methods in complex surface machining required manual gouge checking which is an expensive task. The approach used by Manos *et al.* [12] requires the specification of the tool axis trajectory, not the cutter contact point. The tool axis trajectory fixes two f three coordinates required to specify the cutter location and reduces the problem to solving for one coordinate. In addition, the technique allows for in-built gouge checking. This method doesn't uses traditional method for cutter locations and tool paths as found in previous cutter positioning strategies [1, 13, 16, 22, 10, 4].

In this method, the tool path is generated by separating tool positioning method with tool path foot print based on the surface to be machined. First, a foot print is generated, and this foot print is discretized into number of points where the gouge-free cutter position is calculated. To calculate gouge-free position, the work piece surface is triangularized, i.e. the surface is divided into 3D triangular elements. The tool is then dropped over the triangularized surface of the work piece along the tool axis whose position is specified. Manos *et al.* [12] used a ball nosed tool which has some deficiencies as discussed in Chapter 2. In this work, the method developed by Manos *et al.* [12] is extended to flat and radiused end milling cutters. Here, the ball-nosed, flat and radiused end milling cutters are modeled as a “ball”, a “coin” and a “doughnut” respectively. The cutter is dropped along the tool axis, the first point of contact between the cutter and the surface is selected as it represents a gouge-free cutter location. To find the first point of contact, several cutter locations for different triangles under the tool are calculated and the tool position that represents the highest location of the tool is chosen. To generate the tool path, a series of cutter drops at regularly pre-determined intervals are executed and stored in a file. The file is used to control the depth of cutter as it moves along the specified tool path foot print during machining of a complex surface.

3.1 Surface Data exchange from CAD to CAM

Interoperability among CAD/CAM/CAE systems and the product data communication between solid modeler and CAM systems is a well known problem in product design and development. Data exchange processes from CAD model to CAM systems are usually afflicted by several problems, such as: information loss, redundancy, one-way data exchange and static data exchange. These drawbacks do not permit a really geometric-centric design, and even if the model is transmitted without loss of information, the exchanged data do not incorporate details such as sketches, constraints and features, which represent the designer's intent. As a result, the original intent of the designer may be misunderstood. Thus a streamlined flow of design data that supports all manufacturing processes with precision and ease is desired.

The different modeling philosophies namely Constructive Solid Geometry (CSG), Boundary Representation (B-rep), Solid Modeling was proposed initially to facilitate data exchange. For Solid Modeling, geometrical data exchange among different software packages was proposed through neutral file formats (IGES or STEP) or through proprietary formats. The Initial Graphics Exchange Specification (IGES) file has advantages as it is implemented by almost every commercial CAD/CAM system and also provide the entities of points, lines, arcs, curves, curved surfaces and solid primitives to precisely represent CAD models. Conversely, the IGES file includes much redundant information and it does not support facet representation. Moreover, the algorithms dealing with an IGES file are more complex than those dealing with a STL format [2].

Another format HP/GL (Hewlett-Packard Graphics Language), a standard data format was proposed in late 80's for a graphic plotter. The advantages of using an HP/GL file is that many commercial CAD systems have an interface to output and exchange data in this format, and it can be directly passed to CAM systems for manufacturing. The disadvantage is that many small files are needed to represent a solid object which would result in a management problem which is known to be a cause for machine crashes and will increase computational time.

STEP is a new engineering product data exchange international standard format developed to exchange product data in CAD systems. Since it is a complete representation of a product the information in STEP is complete for data exchange from CAD to CAM systems. It is efficient in both file size and the computer resources needed for processing. STEP is independent of hardware and software. However, STEP still carries far more information then is necessary for CAM systems to generate tool path.

An STL (Stereo Lithography) file which is a *de facto* standard is, an ideal interface between CAD and CAM systems. The STL format is widely accepted for topological data exchange by all CAM systems because of its two major strengths, namely, simplicity and independence from specific CAD modeling methods. Moreover, it can provide an accurate file for data transfer of all shapes. STL will be used in this work because it accurately represents the sculptured surfaces like wooden sign and it does not contains unnecessary information regarding the CAD model. Moreover, STL file can used with any commercial CAD packages for generating tool path.

3.1.1 STL File

An Stereolithography (STL) file is a triangular representation of a 3-dimensional surface geometry. The surface is tessellated or broken down logically into a series of small triangles (facets). Each facet is described by a perpendicular direction and three points representing the vertices (corners) of the triangle.

Format Specifications

An STL file consists of a list of facet data. Each facet is uniquely identified by a unit normal (a line perpendicular to the triangle and with a length of 1.0) and by three vertices (corners). The normal and each vertex are specified by three coordinates each, so there is a total of 12 numbers stored for each facet. There are two basic rules followed for generation of STL file.

1. Facet Orientation

2. Vertex to Vertex Rule

Facet orientation: The facets define the surface of a 3-dimensional object. As such, each facet is part of the boundary between the interior and the exterior of the object. The orientation of the facets (which side is "out" and which side is "in") is specified. First, the direction of the normal specified in the format always points towards the outside of the part. Second, the vertices are listed in counter-clockwise order when looking at the object from the outside (*right-hand rule*). Orientation of a facet is determined by the direction of the unit normal and the order in which the vertices are listed. These rules are illustrated in 3.1.

Vertex to Vertex Rule: Each triangle must share two vertices with each of its adjacent triangles. In other words, a vertex of one triangle cannot lie on the side of another. This is illustrated in Figure 3.2.

The object represented is located in the all-positive octant. In other words all vertex coordinates are positive-definite (non-negative and non-zero) numbers. The STL file does not contain any scale information; the coordinates are in arbitrary units. The official

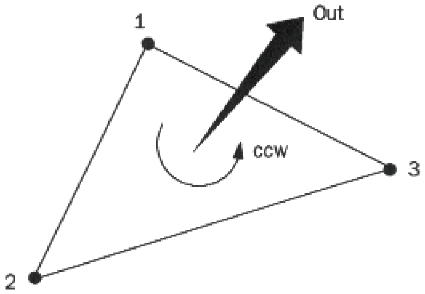
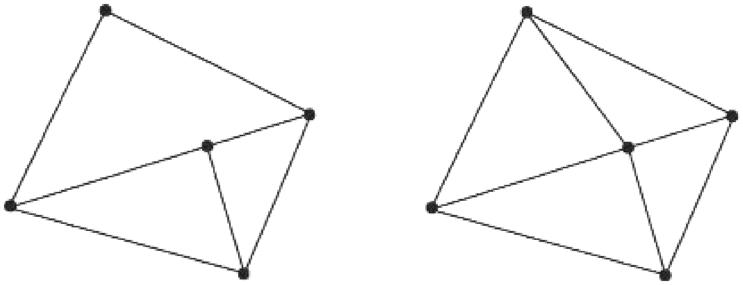


Figure 3.1: Facet Orientation in a STL file



(a) Incorrect Configuration (b) Correct configuration

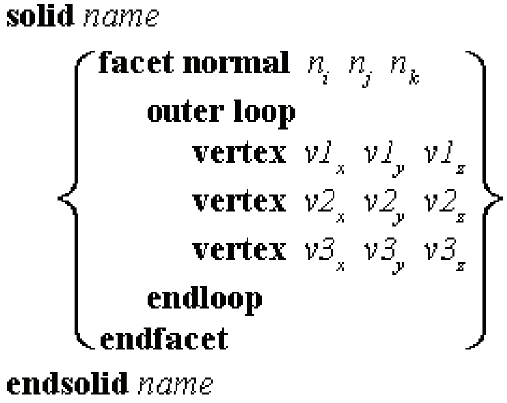
Figure 3.2: Vertex to Vertex Rule [2]

3D Systems STL specification document states that there is a provision for inclusion of special attributes for building parameters, but does not give the format for including such attributes. Also, the document specifies data for the *minimum length of triangle side* and *maximum triangle size*, but these numbers are of dubious meaning. The algorithm may require this extension or it may allow a different extension to be specified. The STL standard includes two data formats, ASCII and binary. These are described separately below.

STL ASCII Format

The ASCII format is primarily intended for testing new CAD interfaces. The large size of its files makes it impractical for general use.

The syntax for an ASCII STL file is as follows:



solid *name*

facet normal n_i n_j n_k

outer loop

vertex $v1_x$ $v1_y$ $v1_z$

vertex $v2_x$ $v2_y$ $v2_z$

vertex $v3_x$ $v3_y$ $v3_z$

endloop

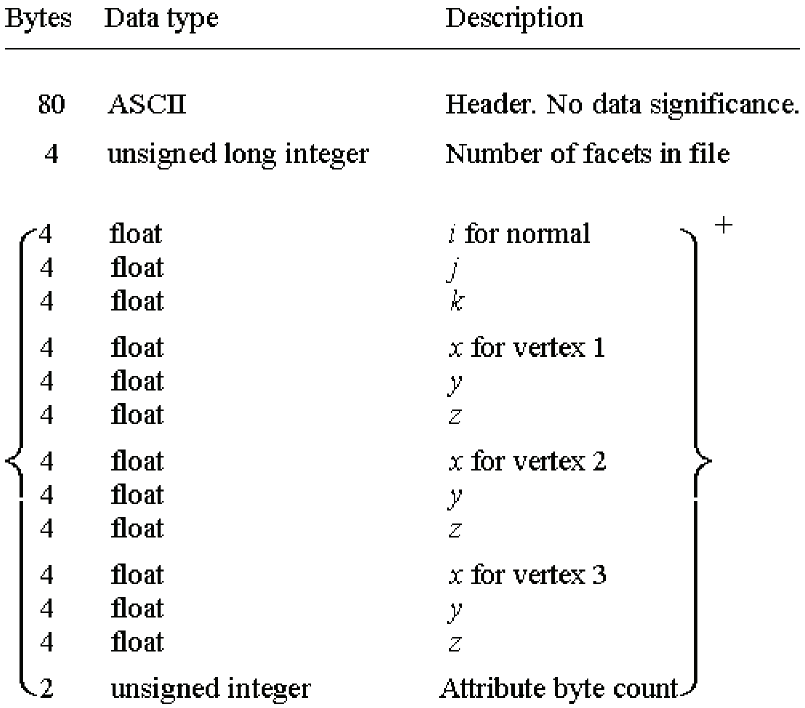
endfacet

endsolid *name*

Bold face indicates a keyword; there is a space in 'facet normal' and in 'outer loop', while there is no space in any of the keywords beginning with 'end'. Indentation must be with spaces; tabs are not allowed. The contents of the brace brackets is repeated one or more times. Symbols in italics are variables which are to be replaced with user-specified values. The numerical data in the facet normal and vertex lines are single precision floats, e.g. 1.23456E+789. A facet normal coordinate may have a leading minus sign; a vertex coordinate may not be negative. The numbers 1, 2 and 3 represent the vertex number i.e. v_1 represent vertex 1, and so on. The subscript x, y and z represents the X , Y and Z coordinate of the vertex. Thus, v_{1x} represents x coordinate of the first vertex of triangle in STL file.

STL Binary Format

The binary format uses the IEEE integer and floating point numerical representation.



Bytes	Data type	Description	
80	ASCII	Header. No data significance.	
4	unsigned long integer	Number of facets in file	
4	float	i for normal	+
4	float	j	
4	float	k	
4	float	x for vertex 1	
4	float	y	
4	float	z	
4	float	x for vertex 2	
4	float	y	
4	float	z	
4	float	x for vertex 3	
4	float	y	
4	float	z	
2	unsigned integer	Attribute byte count	

The notation, + means that the contents of the brace brackets can be repeated one or more times. The content in the braces are repeated as many times as the number of triangles. The attribute syntax is not documented in the formal specification. It is specified that the attribute byte count should be set to zero. Here, The length of the normal vector is 1, vertices follow the right hand rule and the numbers are stored as 32 bit float values. The syntax for a binary STL file is as follows: For generation of tool path the ASCII format of STL file is used.

3.2 Tool path planning method

For the 3 axis milling machines, the cutter is confined to move in and out from the work piece along the confined axis. The cutter is allowed to move in X, Y and Z axis out of which X and Y axis are decided based on the tool path foot print, and the coordinate of Z axis i.e. depth at which the cutter can penetrate is to be found. Each of the cutter depth is combined and sequenced along the specific foot print to generate gouge-free tool path. A tool path is composed of several tool positions based on the radius of the cutter and step-over distance (distance between consecutive cutter positions).

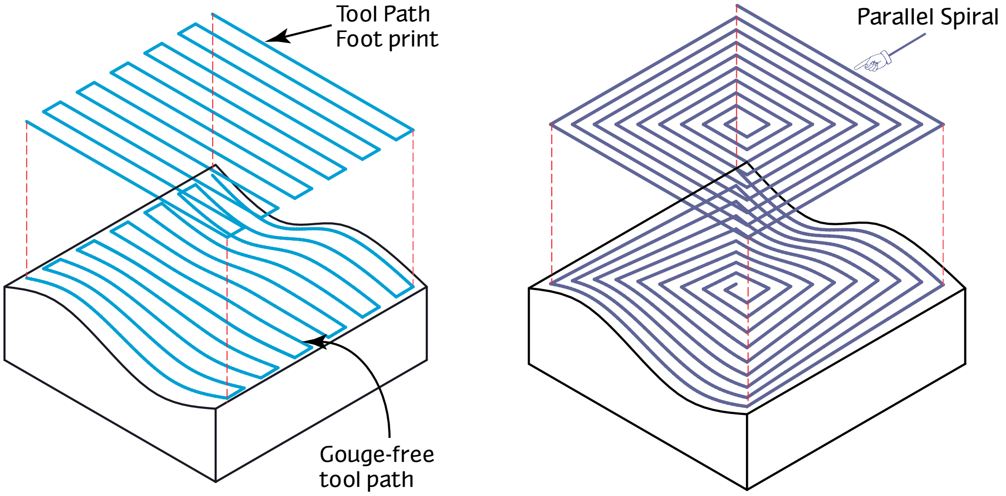
A traditional NC controller can move a tool along any path built up of linear segments. The trajectory of the tool is dictated by a path designed on a plane perpendicular to the tool axis. This trajectory is called the tool path foot print; there is no limit to the shapes of footprints on commercial NC machines. However, it is extremely important to choose optimum footprint because it will affect the final machining time. The footprint is directly related to the tool path length, thus better the footprint lower would be the tool path length.

The path is characterized by its side step i.e. the distance between adjacent straight line paths. To create a tool path, the foot print is discretized into small moves and the "Dropping Method" is used to find the gouge-free cutter position to generate tool path. The tool is moved linearly between these gouge-free positions to machine the part. If the foot print is discretized finely the surface can be machined very precisely. However, time required to generate tool path will increase. The advantage of this method of separating foot print and cutter positioning, is that it can be applied blindly to create the tool path for machining any surface. Moreover, the footprint is generated once, and the same footprint can be used for machining any number of components only the depth will vary at each position.

If the tool path generated can be developed at any X, Y location on the work piece independent of any specific foot print than it will allow generation of custom foot print depending upon the type of the work piece. Thus, by generating fixed topology with an optimal foot print tool path for a family of parts, the mass production of custom products can be achieved. This concept is tested on manufacturing “sculptured wooden sign” which is shown in Section 4.4. Wooden sign are by nature custom. A CAD application that creates infinite combination of shapes, backgrounds, borders, logos and names is used to produce 3D models of signs that are to be machined. This work describes an automatic method for generating tool paths to machine these signs.

Once the foot print is developed, the entire foot print is divided into number of points. The discretization of the foot print into several points is done based on the radius of the cutter and the side-step with which work piece must be machined. The cutter is dropped independent of complexity of the surface at each of these points along foot print and gouge-free cutter location is found. The “Dropping Method” is used to find gouge-free cutter location at each of these points along foot print which is discussed in Section 3.4. By separating the foot print with cutter positioning, the surface can be machined with all type of cutters independent of the complexity of surface. Moreover, the foot print can be customized based on the type of surface to be produced to achieve better surface finish. By machining the surface with flat-end and radiused end mills, more volume of material is removed in single pass which improves machining time.

The method described provides a powerful method for positioning a tool on the part surface; however, a part is machined by moving the tool across the surface. The optimal tool path foot prints for machining a custom sculptured wooden sign with minimum machining time are shown in Figure 3.3a and 3.3b. For majority of rectangular shaped sign with a Zig-zag foot print is developed for a sign which is shown in Figure 3.3a. This foot print can efficiently machine the flat surfaces. For machining more features of a sign in single pass, a different type of foot print was developed namely Parallel spiral foot print which is shown in Figure 3.3b. Border, logo and text are the important features in the sign which needs to be machine these features in a single pass. Thus, a spiral tool path foot print was designed in such a way that it can machine this feature within less time. Once the foot print is designed, all the cutter location is stored in a file containing information regarding footprint and optimal depth for each cutter position.



(a) ZIG-ZAG Foot Print

(b) Parallel Spiral Foot Print

Figure 3.3: Foot print followed by the cutter

3.3 Type of Cutters

By separating tool path planning into foot print and cutter positioning, the tool path can be generated for any type of cutter independent of type of surface being machined. Ball nosed, flat and Radiused end mills are the three most commonly used end-milling cutters in the manufacturing industries. The cutting takes place by rotating cutter about its axis and translating it in a specific direction. As the end milling cutter rotates, its rotational cutting speed is much higher than its translational feed rates. As a result the surface swept by the cutting edges of the ball nosed cutter is modeled as a ball (sphere); for the flat-end milling cutter, it is modeled as coin (cylinder); and for radiused end mill, it is modeled as a torus (doughnut). The geometry of each of the three milling cutters is described in Figure 3.4. The radiused end mills are generally produced by filleting the bottom edge of a flat-end mill with a desired radius. A Radiused end mills can represent the shape of both ball nose and flat-end mill i.e. circular insert as minor radius (R_1) and flat portion with major radius (R_2). Thus, if any $R_1 = 0$ then the geometry of the cutter will be same as flat-end milling cutter and, if $R_2 = 0$ then it would resemble ball nosed cutter.

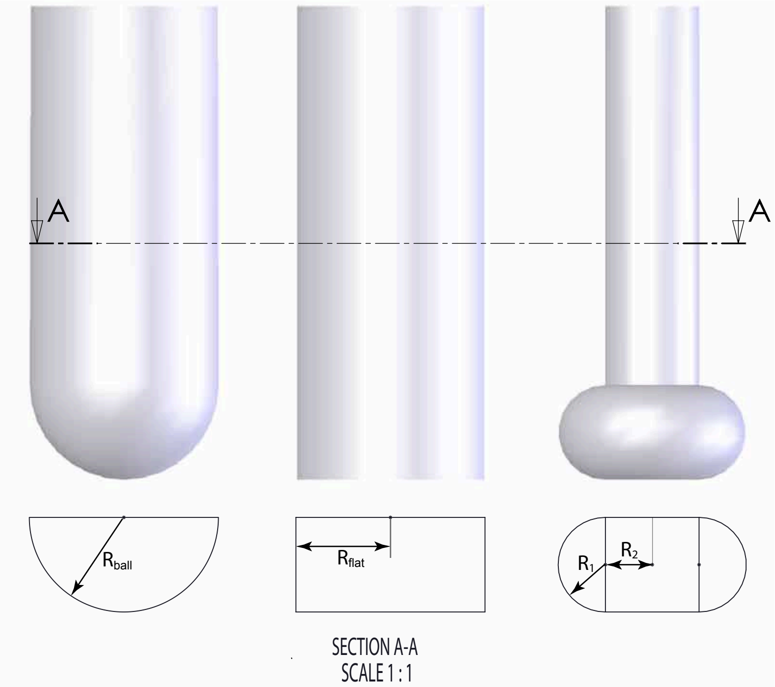


Figure 3.4: Geometry of Ball nose, Flat and Radiused End Milling Cutters

The ball nosed end mill has a fixed circular cutting surface, this will lead to larger scallop heights and require more passes to machine a part. Moreover, the tip of the ball has zero radius cutting speed, which leads to tearing of the work piece material resulting in poor surface finish. The flat-end mills results in minimum scallop heights as there is surface contact instead of point contact. The zero corner radius for flat-end mills would produce marks along the feed direction which increases surface roughness. A toroidal cutter inherits the merits of both ball nose and flat-end milling cutters, as it do not have zero corner radius or zero radius tip [18]. The tool path planning methodology named "Dropping Method" has been developed for each of the three milling cutters. The procedure for calculating gouge-free cutter position for each of the three end milling cutter and developing optimal tool path is discussed in Section 3.4.

3.4 "Dropping Method" for Gouge-free Cutter Position

In "Dropping Method", for each cutter position the tool is dropped from a height along the tool axis at any particular position in $X - Y$ plane. The cutter drops steadily along tool axis plane and would intersect one of the triangles representing the work piece surface. There are three possible scenarios for the point of contact between cutter and surface: 1) contact on planar surface, 2) contact on edge, and 3) contact on vertex. The point of contact will be dependent of the topography of the surface. A brief description of the checks is given in Figure 2.2.

The steps to determine the cutter depth at which it contacts the triangles are:

1. Shadow Check

2. Triangles Check

3. Edges Check
4. Vertex Check

The flow diagram for the procedure of selecting optimal cutter location is shown in Figure 3.5. As shown in Figure 3.4 the ball nose cutter is defined a ball of radius R, the flat-end mill is defined with a cylinder of radius R, and the radiused end mill is defined with two radii i.e. insert of radius R_1 and the offset of the insert by radius R_2.

3.4.1 Shadow Check

Shadow check is performed to find the triangles in the "shadow" of the tool at a certain location. If the cutter were to plunge into the material the triangles that are removed would be considered to lie in the shadow of the tool. Thus, for any particular cutter location all the triangles in the surface that are within the limits of the "shadow" of the tool are found. The remaining three checks will only be applied to the triangles in the tool shadow. This results in faster processing times since the number of checks is less. The diagrammatic representation of Shadow check is shown in Figure 3.6, where shaded triangles indicates the triangle under the shadow of the cutter.

The X and Y coordinate of cutter position is known. Thus for calculating the shadow, the geometry of the cutter is projected on the XY-plane. This geometry is made independent of Z axis, which will cause the shadow to plunge completely through the work piece. Thus, a shadow is a 2-dimensional circle in XY plane and all the triangles are checked for three special cases for shadow check shown in Figure 3.7, and a triangle that satisfies any of the cases is considered to be in the shadow of the cutter.

Here, for case 1 Figure 3.7a a triangle's vertex is checked whether it is inside the circle or

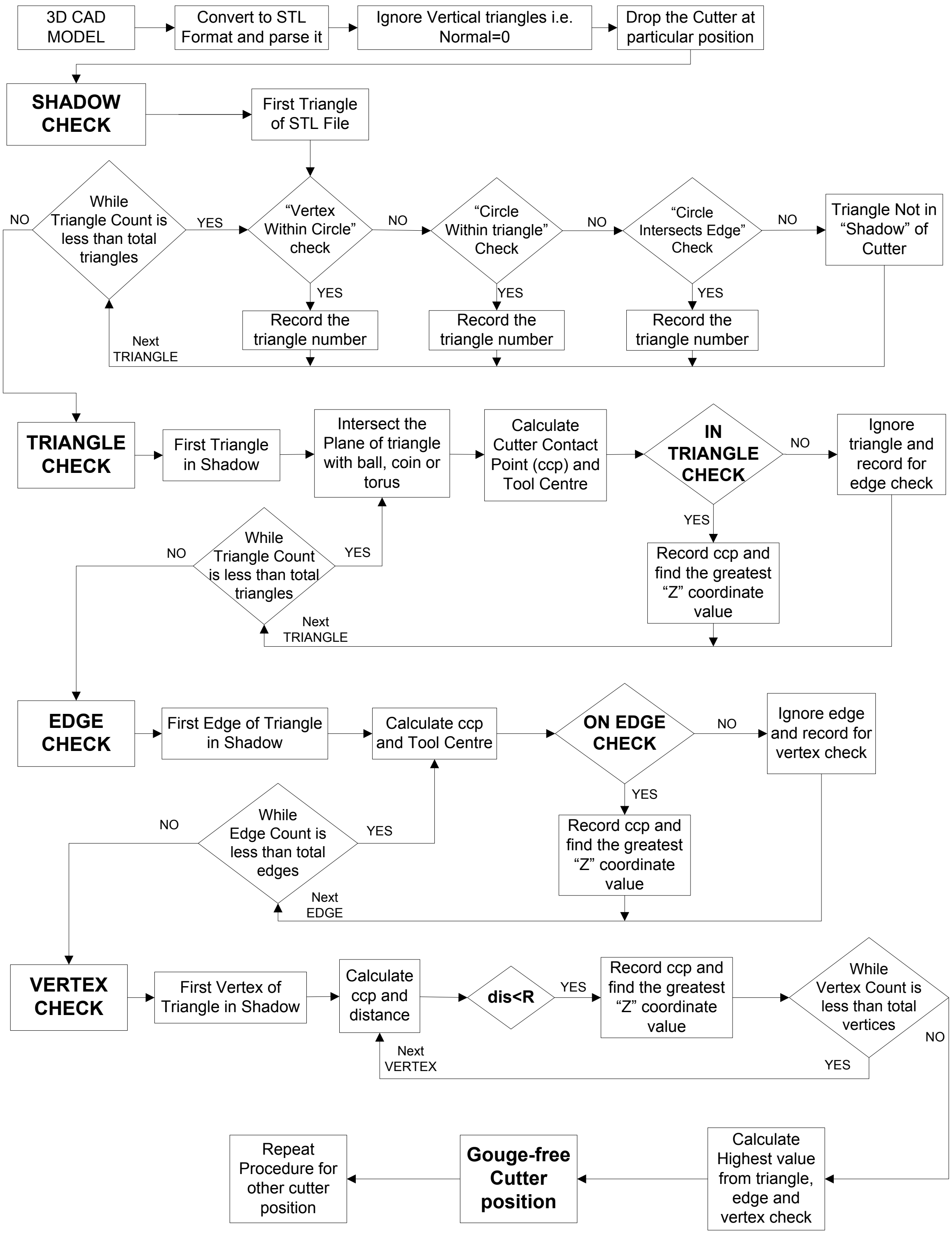

3D CAD MODEL
Convert to STL Format and parse it
Ignore Vertical triangles i.e. Normal=0
Drop the Cutter at particular position
SHADOW CHECK
First Triangle of STL File
NO
While Triangle Count is less than total triangles
YES
"Vertex Within Circle" check
NO
"Circle Within triangle" Check
NO
"Circle Intersects Edge" Check
NO
Triangle Not in "Shadow" of Cutter
YES
YES
YES
Record the triangle number
Record the triangle number
Record the triangle number
Next TRIANGLE
TRIANGLE CHECK
First Triangle in Shadow
Intersect the Plane of triangle with ball, coin or torus
Calculate Cutter Contact Point (ccp) and Tool Centre
IN TRIANGLE CHECK
NO
Ignore triangle and record for edge check
YES
Record ccp and find the greatest "Z" coordinate value
NO
While Triangle Count is less than total triangles
YES
Next TRIANGLE
EDGE CHECK
First Edge of Triangle in Shadow
Calculate ccp and Tool Centre
ON EDGE CHECK
NO
Ignore edge and record for vertex check
YES
Record ccp and find the greatest "Z" coordinate value
NO
While Edge Count is less than total edges
YES
Next EDGE
VERTEX CHECK
First Vertex of Triangle in Shadow
Calculate ccp and distance
dis<R
YES
Record ccp and find the greatest "Z" coordinate value
While Vertex Count is less than total vertices
NO
YES
Next VERTEX
Calculate Highest value from triangle, edge and vertex check
Gouge-free Cutter position
Repeat Procedure for other cutter position

Figure 3.5: Flow Diagram for selection of gouge free cutter location

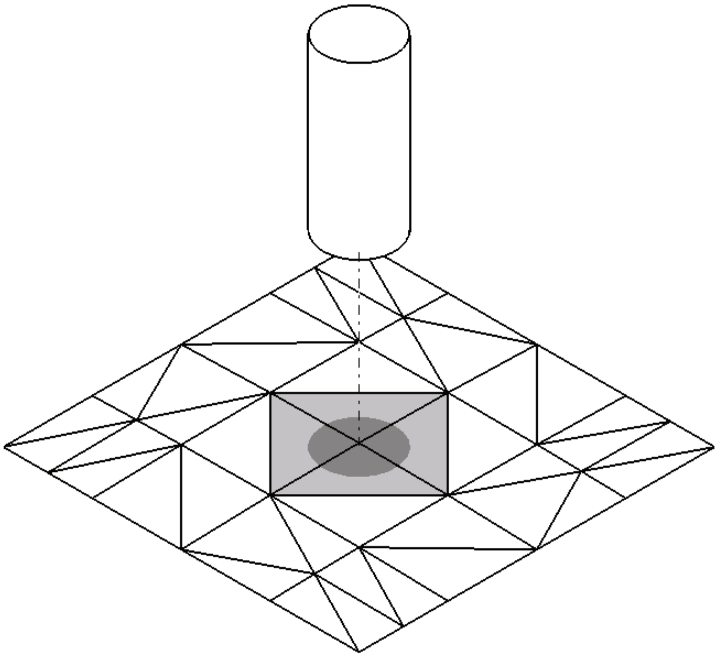


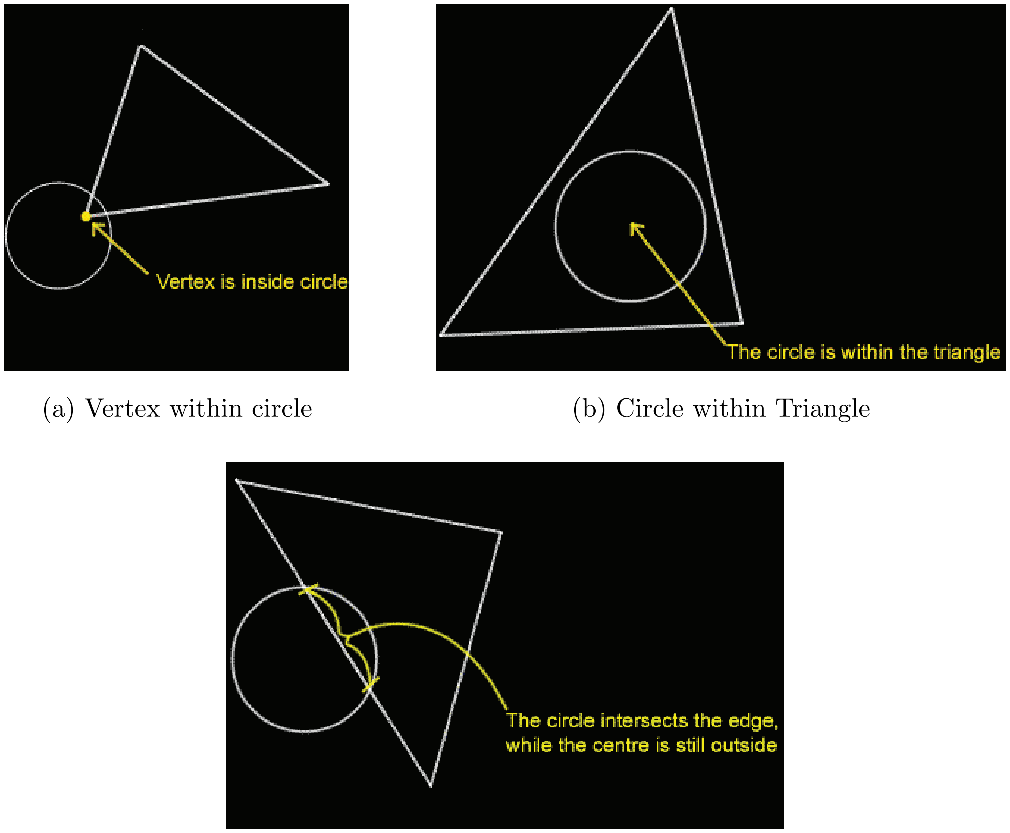
Figure 3.6: Estimation of triangles under “shadow” of the cutter

not. The distance between each vertex of the triangle and the origin at the circle’s centre is calculated in X and Y direction. If the length is larger than radius of the cutter than the triangle may be outside the shadow or it is considered in the shadow and other tests are conducted. This check is repeated for all the three vertices of the triangle.

$$\vec{k_1} = \vec{v_1} - \vec{T_c}; \quad \vec{k_2} = \vec{v_2} - \vec{T_c}; \quad \vec{k_3} = \vec{v_3} - \vec{T_c} \tag{3.1}$$

Here, $\vec{v_1}$, $\vec{v_2}$, $\vec{v_3}$ and $\vec{T_c}$ represent vertices of triangle and Tool Centre respectively. The condition for checking this case is

```
if $sqrt(\vec{k_i} \bullet \vec{k_i}) \leq Radius$ then
    Triangle Inside the Shadow
else
    Triangle Outside the Shadow
end ifSimilarly, Repeated for the other vertices with $i = 2$ and $i = 3$
```


Vertex is inside circle
The circle is within the triangle
The circle intersects the edge,
while the centre is still outside

(a) Vertex within circle

(b) Circle within Triangle

(c) Circle Intersects Edge

Figure 3.7: Cases to check whether triangle is inside the shadow

For case 2 shown in Figure 3.7b, the circle doesn't intersect any of the triangle's vertices, A test that checks whether the circle is inside the triangle or not is performed. In this check the centre of the circle checked to see if it is on the positive side of all edges that make up the triangle. This condition is true only when the point tested (centre) is within the triangle. This condition is satisfied even if a line is infinite, because all three of them must be tested and the point has to be in front of **ALL** to actually be within the triangle. It must be kept in mind that the lines should be defined in such a way that the inside of the triangle is on its positive side. This is schematically shown in Figure 3.8b. The Normal

Vector and Edge vectors for different edges of the triangles are given by

$$\vec{c_1} = \vec{T_c} - \vec{v_1}; \quad \vec{c_2} = \vec{T_c} - \vec{v_2}; \quad \vec{c_3} = \vec{T_c} - \vec{v_3} \tag{3.2}$$

$$\vec{N_1} = \vec{v_2} - \vec{v_1}; \quad \vec{N_2} = \vec{v_3} - \vec{v_2}; \quad \vec{N_3} = \vec{v_1} - \vec{v_3} \tag{3.3}$$

The Pseudo-code for this case is shown algorithmically below

```
if $\vec{N_1} \bullet \vec{c_1} \geq 0$ and $\vec{N_2} \bullet \vec{c_2} \geq 0$ and $\vec{N_3} \bullet \vec{c_3} \geq 0$ then
    Triangle Inside the Shadow
end if
```

For case 3 shown in Figure 3.7c a check is made to see if the circle intersects an edge of the triangle. If the shortest distance from the centre of the circle to the edge is within the circle's radius, than the circle intersects the edge at two unique points. The perpendicular from a point to a line is the shortest distance. Once the intersection point is found it is checked whether the intersection point is on the edge and if the intersection point lies beyond the bounding vertices than it is neglected.

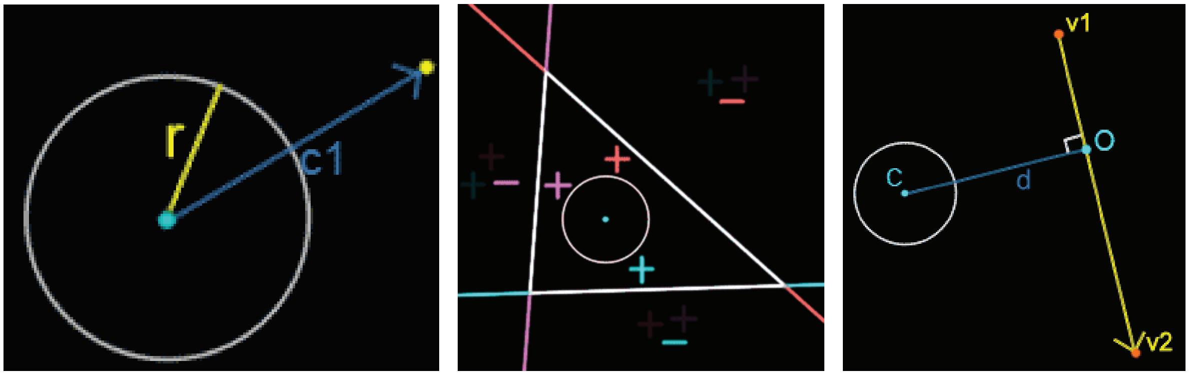
$$\vec{e_1} = \vec{v_2} - \vec{v_1}; \quad \vec{e_2} = \vec{v_3} - \vec{v_2}; \quad \vec{e_3} = \vec{v_1} - \vec{v_3} \tag{3.4}$$

The magnitude of the edges of the triangles are calculated using Equation 3.5

$$length = \sqrt{\vec{e_i} \bullet \vec{e_i}}, \quad i = 1 \cdots 3 \tag{3.5}$$

The Pseudo-code for this case is shown algorithmically below

```
if $\vec{c_i} \bullet \vec{e_i} > 0$ then
    if $\vec{c_i} \bullet \vec{e_i} < length$ then
        if $(\vec{c_i} \bullet \vec{c_i}) - (\vec{c_i} \bullet \vec{e_i}) \leq R^2$ then
            Triangle Inside the Shadow
        end if
            end if
        end if Similarly, Repeated for the edges with $i = 2$ and $i = 3$
```



(a) Vertex within circle (b) Circle within Triangle (c) Circle Intersects Edge

Figure 3.8: Geometry Analysis of Cases for Shadow Check

The three checks are done on all the triangles that make up a part and the triangles that lie in the shadow are selected for further processing.

3.4.2 Triangle Check

During this stage each triangle in the shadow of the cutter is considered for further checks and other triangles are neglected. The purpose to this check is to determine whether a tool of radius R moving along tool axis will contact a triangle. The cutter location is calculated such that the tool contacts the plane of triangle tangentially. Based on the geometry, two solutions exists- one where the cutter is located along the positive side of the normal (pointing outwards) and other where the cutter is located on the negative side of the normal. The solution which represents the contact with the positive side of the normal cutter is accepted and other is rejected.

Here, V_1, V_2 and V_3 represent vertices of the triangle. L_1 represent vertex of the triangle, whereas $\vec{L_2}$ and $\vec{L_3}$ represent the direction vector along the second and third edge of the triangle.

$$L_1 = (V_{1x}, V_{1y}, V_{1z}) \tag{3.6}$$

$$\vec{L_2} = \frac{V_2 - V_1}{|V_2 - V_1|} = \left(\frac{V_{2x} - V_{1x}}{|V_2 - V_1|}, \frac{V_{2y} - V_{1y}}{|V_2 - V_1|}, \frac{V_{2z} - V_{1z}}{|V_2 - V_1|} \right) \tag{3.7}$$

$$\vec{L_3} = \frac{V_3 - V_1}{|V_3 - V_1|} = \left(\frac{V_{3x} - V_{1x}}{|V_3 - V_1|}, \frac{V_{3y} - V_{1y}}{|V_3 - V_1|}, \frac{V_{3z} - V_{1z}}{|V_3 - V_1|} \right) \tag{3.8}$$

The standard equation of a plane in 3 space with non-zero normal vector $\vec{n} = (\vec{n_x}, \vec{n_y}, \vec{n_z})$ is shown in Equation 3.9.

$$\vec{n_x}X + \vec{n_y}Y + \vec{n_z}Z + d = 0 \tag{3.9}$$

The general equation of plane formed by three vertices of the triangle V_1, V_2 and V_3, and passing through these vertices is given by the following determinants. The sign of the Equation 3.9 determines on which side the vertex lies with respect to the plane. If sign is positive than vertex lies on the same side as the normal, and if sign is negative than vice-versa.

$$\vec{n_x} = \begin{vmatrix} 1 & V_{1y} & V_{1z} \\ 1 & V_{2y} & V_{2z} \\ 1 & V_{3y} & V_{3z} \end{vmatrix} \vec{n_y} = \begin{vmatrix} V_{1x} & 1 & V_{1z} \\ V_{2x} & 1 & V_{2z} \\ V_{3x} & 1 & V_{3z} \end{vmatrix} \vec{n_z} = \begin{vmatrix} V_{1x} & V_{1y} & 1 \\ V_{2x} & V_{2y} & 1 \\ V_{3x} & V_{3y} & 1 \end{vmatrix} d = - \begin{vmatrix} V_{1x} & V_{1y} & V_{1z} \\ V_{2x} & V_{2y} & V_{2z} \\ V_{3x} & V_{3y} & V_{3z} \end{vmatrix} \tag{3.10}$$

In this stage all the triangles in the shadow of the cutter are considered and each triangle is represented as a unique plane. Since the centre of the cutter lies along the tool axis, the cutter area contacts the plane of the triangle tangentially. Equation 3.32, 3.12 and 3.13 represents geometry of the plane contacting the ball nose, flat and radiused end milling cutters respectively.

Ball Nose End mill

$$T_1 + u\vec{T_2} = L_1 + t\vec{L_2} + s\vec{L_3} + \hat{N}R \tag{3.11}$$

Flat-End mill

$$T_1 + u\vec{T_2} = L_1 + t\vec{L_2} + s\vec{L_3} + \hat{N}_{proj}R \tag{3.12}$$

Radiused End mill

$$T_1 + u\vec{T_2} = L_1 + t\vec{L_2} + s\vec{L_3} + \hat{N}R_1 + \hat{N}_{proj}R_2 \tag{3.13}$$

where,

$$\hat{N} = \frac{\vec{L_2} \times \vec{L_3}}{|\vec{L_2} \times \vec{L_3}|}$$

and

$$\hat{N}_{proj} = \hat{N} - (\hat{N} \cdot T)T$$

Here t, s and u represent the parameters for vector $\vec{L_2}$, $\vec{L_3}$ and tool axis respectively. Solving the result obtained from above equations for each type of tool, the contact point P_c between the tool and the plane of the triangle, and the location of tool T_c along the tool axis is calculated.

In-Triangle Check

The potential tool centre determined by the "Triangle Check" must be checked to see if it lies within the boundaries of the triangle. If the point lies within the boundaries of the triangle, the cutter location is recorded otherwise it is discarded and the next check is performed. This task is accomplished by using a Barycentric Technique, most efficient and simple in terms of calculation.

In this technique, one of the vertices of the triangle is chosen and all other locations on the plane are defined relative to that point. The newly formed coordinates of the vertices of the triangle are called Barycentric coordinates. Now choose any vertex of the given triangle as the base vertex and the other vertices are calculated from that. So a point within the triangle can be reached by starting at V_1 and walking some distance along $(V_2 - V_1)$ and then walking along in the direction (V_3-V_1). The detailed explanation is given in a Figure 3.9

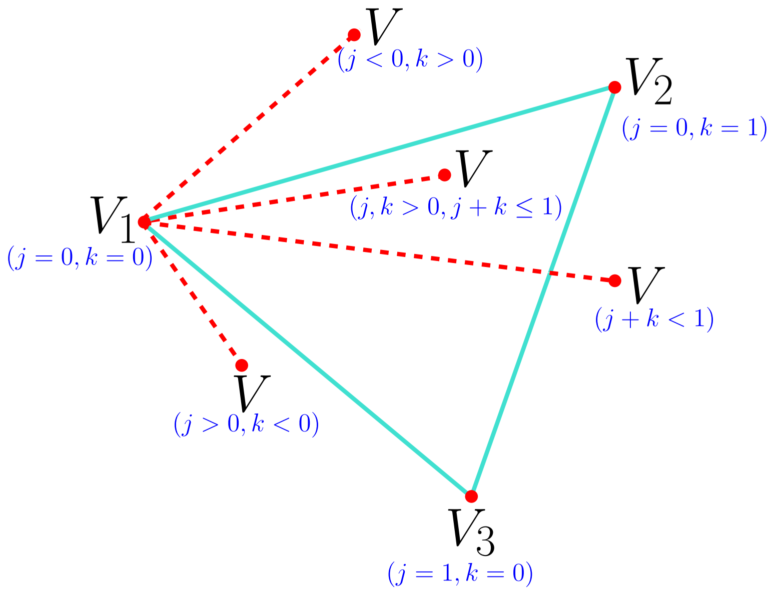


Figure 3.9: Barycentric technique for In-triangle Check

Therefore any point in the plane can be described by the expression

$$V = V_1 + j * (V_2 - V_1) + k * (V_3 - V_1) \tag{3.14}$$

where j and k are vectors along the edge $\overline{V_1V_2}$ and $\overline{V_1V_3}$ respectively. Now the above expression is solved for j and k, which will decide whether the ccp is inside the triangle or not. Few important results can be obtained from the values of j and k. If $j = 1$ & $k = 0$

or $j = 0\,\&\,k = 1$ or $j = 0\,\&\,k = 0$ than it implies that ccp is on vertex V_2, V_3 and V_1 respectively. If $j\ or\ k < 0$ then ccp is in the wrong direction and is outside the triangle. Also if $j\ or\ k > 1$ then ccp is too far in a direction and is outside the triangle. Finally if $j + k > 1$ then the point has crossed the edge $\overline{V_2V_3}$ and it is outside the triangle.

Thus, if $j >= 0\ and\ k >= 0\ and\ j + k <= 1$ then the ccp and tool centre is recorded, otherwise it can be deduced that ccp lies outside the lines that define triangle.

3.4.3 Edge Check

This check is done to find out if the tool will touch an edge tangentially within the limits of the vertices defining the edge. The edge check is carried out for each of the edges in the triangle. First, it must be determined at what point the tool touches the edge. The contact location P_c will generate two solutions, but the CL that yields greatest distance from work piece axis is chosen. The ccp is then checked to see if it lies within the endpoints. If the ccp is within the edge the cutter location is saved, otherwise it is discarded and the final check is performed.

Firstly, the shortest distance *gap* between the tool axis $T(u) = T_1 + u\vec{T_2}$ and the original edge of the triangle $L(s) = L_1 + s\vec{L_2}$ is calculated. The check is performed only if $gap < R$, otherwise the edge is neglected. The shortest distance between edge line segment and tool axis is calculated as $gap = \vec{L_2} \cdot \vec{T_2}$. Once, the *gap* is checked, the edges are divided into two separate categories edge perpendicular to the tool axis and edge inclined to the tool axis. If the edge is perpendicular to the tool axis then the calculation of ccp is simplified otherwise, the ccp is calculated through vector geometry between cutter and edge of the triangle. In the section below the geometry formed by contacting the edge of the triangle with the ball nose, flat and radiused end milling cutters is described. Figure 3.10 shows the vector diagram of the geometry formed by ball nose and radiused end milling cutter.

Ball Nose End Mill

From the geometry of the ball nose cutter with the edge of the surface, the first contact point is calculated. If the edge lies in the $X - Y$ plane (i.e. perpendicular to tool axis) then, s and u could be easily calculated using the equation 3.18. This equation will result in 3 equations and 2 unknown u and s. For the cases where the edge is perpendicular to the tool axis, the parameter $u = u_A$ and $s = s_A$. The parameters are calculated only if the edge satisfies the condition $gap \leq R$, otherwise the edge is discarded. The highest value of contact point with the edge from all the triangles is stored in an array.

$$T_1 + u_A\vec{T_2} = L_1 + s_A\vec{L_2} + \hat{N}gap \tag{3.15}$$

where,

$$\hat{N} = \frac{\vec{L_2} \times \vec{T_2}}{|\vec{L_2} \times \vec{T_2}|}$$

For the edges that are inclined to the tool axis, the ccp is calculated through vector geometry formed between cutter and workpiece as shown in Figure 3.10a. Here, from the geometry it can be seen that $\overline{T_C L_N}$ is perpendicular to $\overline{L_A L_N}$. Here, u_A and s_A represent the parameter values of the point $\overline{T_A}$ and $\overline{L_A}$. The points $\overline{T_A} = T_1 + u_A\vec{T_2}$ and $\overline{L_A} = L_1 + s_A\vec{L_2}$ are calculated by solving Equation 3.18. Now, from the vector geometry shown in Figure 3.10a, Equation 3.16 and 3.17 are obtained.

$$(\overline{T_C} - \overline{L_N}) \cdot (\overline{L_A} - \overline{L_N}) = 0 \tag{3.16}$$

$$(\overline{L_N} - \overline{L_A})^2 + \sqrt{gap^2 + (\overline{T_C} - \overline{T_A})^2} = R^2 \tag{3.17}$$

Solving the above equations would result in a quadratic equation in terms of s. The closed form solutions for the parameters s and u in Equation 3.16 and 3.17, is implemented into the computer program. From the values of s and u the tool centre location and gouge-free cutter contact point are subsequently calculated.

Flat-End Mill

The Equation for edge check for flat-end mill is similar to Equation 3.18 for ball nose cutter except the normal vector $\hat{N}$ is $\hat{N_{proj}}$.

$$T_1 + u\vec{T_2} = L_1 + s\vec{L_2} + \hat{N_{proj}} gap \tag{3.18}$$

From the above equation s and u is calculated.

Radiused End Mill

The vector geometry of radiused end mill for edge check calculation is shown in Figure 3.10b. Based on the geometry formed between radiused end milling cutter and the edge of the triangle following equations are found.

$$T = T_1 + u\vec{T_2} \tag{3.19}$$

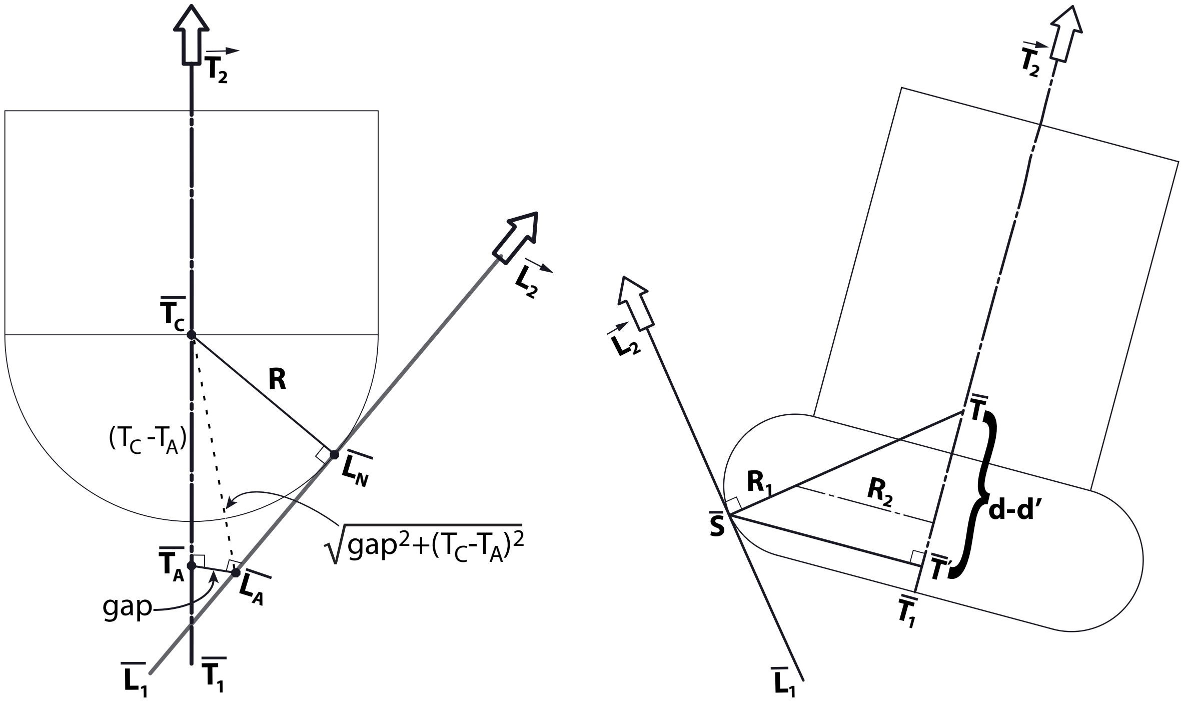
$$S = L_1 + s\vec{L_2} \tag{3.20}$$

$$T^{'} = T_1 + u^{'}\vec{T_2} \tag{3.21}$$

From the geometry of radiused end milling cutter and the edge in the Figure 3.10b, it can be seen that $\overline{TS}$ is perpendicular to edge direction vector $\vec{L_2}$, and $\overline{T'S}$ is perpendicular to tool axis $\vec{T_2}$. Thus, it results in

$$(T - S) \cdot \vec{L_2} = 0 \tag{3.22}$$

$$(T^{'} - S) \cdot \vec{L_2} = 0 \tag{3.23}$$



(a) Ball Nose End Mill

(b) Radiused End mill

Figure 3.10: Vector Geometry of Edge of the triangle for Ball Nose and Radiused end mill

Solving the Equation 3.22 and Equation 3.23, Equation 3.24 and Equation 3.25 can be obtained respectively.

$$u = \left(\frac{\vec{L_2}\cdot\vec{L_2}}{\vec{T_2}\cdot\vec{L_2}}\right)s + \left(\frac{\vec{L_1}\cdot\vec{L_2} - \vec{T_1}\cdot\vec{L_2}}{\vec{T_2}\cdot\vec{L_2}}\right) = a_{11}s + a_{10} \tag{3.24}$$

$$u^{'} = \left(\frac{\vec{L_2}\cdot\vec{T_2}}{\vec{T_2}\cdot\vec{T_2}}\right)s + \left(\frac{\vec{L_1}\cdot\vec{T_2} - \vec{T_1}\cdot\vec{T_2}}{\vec{T_2}\cdot\vec{T_2}}\right) = a_{21}s + a_{20} \tag{3.25}$$

where a_{11}, a_{10}, a_{21} and a_{20} are the constants in the Equation. The A^2 represents the distance $\left|T^{'} - S\right|$ as seen in the Figure 3.10b. The distance is given by

$$A^2 = \left|T^{'} - S\right| = \left(T^{'} - S\right)\cdot\left(T^{'} - S\right)$$

$$A^2 = \left((T_1 - L_1) + u^{'}\vec{T_2} - s\vec{L_2}\right) \cdot \left((T_1 - L_1) + u^{'}\vec{T_2} - s\vec{L_2}\right) \tag{3.26}$$

Now all the values are collected in terms of s and finally Equation 3.27 is obtained. The Equation 3.27 is a Second Degree Polynomial Equation of A in terms of s.

$$\begin{aligned} A^2 \;=\;& s^2 \left[\vec{L_2} \cdot \vec{L_2} - a_{21}\left(\vec{L_2} \cdot \vec{T_2}\right)\right] \\ & -s\left[2\vec{L_2}\left(T_1 - L_1\right) + a_{20}\vec{L_2}\vec{T_2} - a_{21}\vec{T_2}\left(T_1 - L_1\right)\right] \\ & +\left[a_{20}\vec{T_2}\left(T_1 - L_1\right) + \left(T_1 - L_1\right)\left(T_1 - L_1\right)\right] \end{aligned} \tag{3.27}$$

The Geometry of the doughnut-shaped cutter and the edge of the triangle representing work piece gives the following relationship,

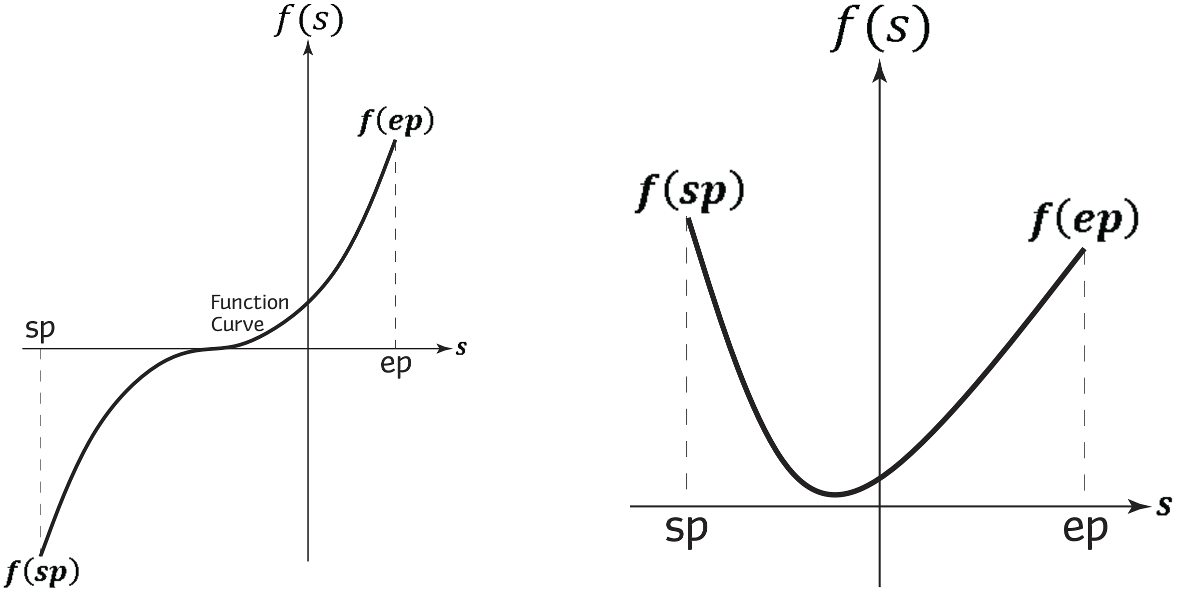
$$R_1 \cos\theta + R_2 = T^{'} - S$$

$$R_1\left(\frac{A}{\sqrt{A^2 + \left(u - u'\right)^2}}\right) + R_2 = A \tag{3.28}$$

$$R_1 A + R_2\sqrt{A^2 + \left(u - u'\right)^2} = A\sqrt{A^2 + \left(u - u'\right)^2} \tag{3.29}$$

The value of A in terms of s from Equation 3.27 is placed in Equation 3.29. This will result in an ***Eighth Order Equation*** in terms of s, u and $u^{'}$. Solving this Eighth Order Equation fetches the value of the parameters s, u and $u^{'}$ which ultimately results in gouge-free ccp and Tool Centre.

The roots of the eighth order equation are solved using the Bisection Method. The bisection method is a root-finding algorithm which repeatedly bisects an interval then selects a subinterval in which a root must lie for further processing. In order to find roots of the eight order equation, the function $f(s)$ is first evaluated at equally spaced intervals of s until two successive function values are found with opposite signs. The bisection method works only when the initial interval of uncertainty (sp, ep) contains an odd number of roots as shown in Figure 3.11a. This method will not work if the interval (sp, ep) contains a double root (as in Figure 3.11b), since $f(sp)$ and $f(ep)$ will have the same sign. For the cases mentioned in Figure 3.11b, the roots of the equation are found through geometrical calculations between cutter and work piece.



(a) Opposite signs of function at start point (sp) and end point (ep)

(b) Same signs of function at start point (sp) and end point (ep)

Figure 3.11: Cases to solve Eighth order Equation

The closed form solution of s and u is obtained by solving Equation 3.29. Once the closed form solution of s is calculated from eighth order equation, the ccp and the tool centre is calculated. More than one solution for the parameters s and u generally exist. The depth of the cutter for each case is found by substituting back the parameters into the equation to find tool centre.

$$TC = T_1 + u\vec{T_2}$$

The value of tool centre is temporarily stored in an array. The tool centre for all the edges in the shadow of the current cutter position is found. The largest resulting tool centre is chosen and it represents the first point of contact with the edge for particular tool position.

On-Edge Check

Once the point of contact and tool centre is calculated, a check is performed to see whether point of contact is on the edge of the triangle. A check is made on the coordinates of ccp before recording the cutter location. If the value of s of the contact point is between 0 and 1 then the ccp lies on the edge. As the x and y coordinates of the contact point are bounded by the two vertices forming the edge. This insures that the point being machined is part of the approximated surface and not a point off in 3D space.

3.4.4 Vertex Check

The final check for the intersection with the vertices of the triangle is performed. The point of contact of the tool with each vertex is found by solving the equation below for u. The solution of the equation will yield two unique values for u. If both the solutions are non-real values then it mean that the tool does not touch vertex V_i for any value of u. In the case of two real value solutions the one furthest away from the work piece is taken as the cutter location.

The approach for the vertex is similar to the edge check except here the single point is considered. To determine tool centre, the distance between the tool axis and the vertex of the triangle is checked. This check is performed to see if a vertex L_1 will touch the cutter as it moves along the line tool axis. From the Geometry of the cutter shown in Figure

3.12, the shortest distance is calculated. For the ball nose and flat-end mills the check is performed only if the distance between cutter and the vertex *dist* is less than radius of the cutter. For the radiused cutter the ccp is calculated for two different cases i.e. $dist \leq R_2$ and $dist < R_1 + R_2$ shown in Equation 3.33 and 3.34. Based on the case u, ccp and cutter location are calculated. The check is performed for all three vertices of the triangle in the shadow of the cutter.

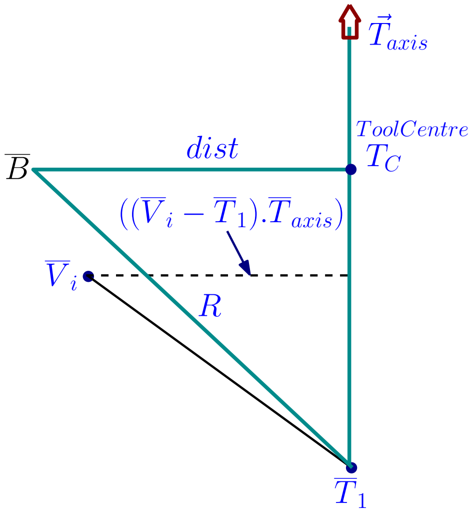


Figure 3.12: Vector diagram for Vertex Check

The distance between the tool axis $T_{axis} = \overline{T_1} + u\vec{T_2}$ is calculated from the geometry formed with vertex of the triangle. The $\overline{B}$ is the point formed between tool centre and the vertex of the triangle.

$$\overline{B} = (\overline{V}_i - \overline{T}_1) - [((\overline{V}_i - \overline{T}_1) \cdot \overline{T}_{axis})\overline{T}_{axis}] \qquad i = 1 \ldots 3 \tag{3.30}$$

The magnitude of the $\overline{B}$ is the shortest distance *dist* between a vertex of the triangle and tool axis.

$$dist = \sqrt{B(0)^2 + B(1)^2 + B(2)^2} \tag{3.31}$$

The procedure for solving *dist* is same for all the three cutters. The tool centre parameter u is calculated differently for ball nose and radiused cutters.

Ball Nose End mill

$$u = ((\overline{V}_i - \overline{T}_1).\overline{T}_{axis}) + \sqrt{R^2 - dist^2} \tag{3.32}$$

Radiused End mill

The tool centre parameter for two separate cases when $dist \leq R_2$ and $dist < R_1 + R_2$ is shown below.

$$u = ((\overline{V}_i - \overline{T}_1) + R_1 \tag{3.33}$$

$$u = ((\overline{V}_i - \overline{T}_1) + R_1 \sin\theta \tag{3.34}$$

and the angle θ is calculated from the geometry between the tool centre and triangle.

$$\cos\theta = (dist - R_2)/R_1$$

The value of tool centre parameter is placed in a equation $TC = T_1 + u\vec{T_2}$ and a gouge-free tool position is found. Once the three checks have been completed the resulting cutter locations are scanned in order to find the cutter location that is closest to the height from which cutter is dropped. The resulting cutter location represents the point at which the cutter will touch the surface without gouging the surface. This procedure is than repeated for all other tool position and gouge-free tool-path is generated.

3.5 Web-based tool path planning

For developing full fledged web-based application for customized products, it is necessary to automate the CAM system and allow users to generate G codes for machining if the customer demands. Personalized wooden signs were chosen as a mass customized product for this project. The goal of this system is to allow users to design and manufacture customized wooden signs using a web-based interface. This CAM system is merged with a web-based design application developed by Kandala [21] for manufacturing wooden signs. Here users were allowed to design their own logos, add personalized text anywhere on the sign and select border and background from standard designs. Once the user finishes the design, the 3D CAD model of the sign is visualized over the web browser and the user can make their tool path and machine their custom sign. Upon selection, a CAM application is initiated by CAD Macros over the web browser. For making the tool path, few basic information regarding the type of machine, type of tooling, type of footprint and output file type is taken from the user. Thus, the web-based CAM Application would generate tool path that is compatible with their machine so the users just has to load the tool path on the machine to manufacture a sign. Similarly for machining a custom sign for user, the standard machine and cutters are chosen to manufacture a sign. The tool paths are checked in a simulator and the video is shown over the web browser to the user.

CAD/CAM Communication

There is transfer of information from client web application to the server side Solid modeler for creating STL file. This application was built on ASP.NET for transferring information between server and client. The scripts on the server side are hidden behind the VB.NET code stored in *Default.aspx.vb* for the web page. The core tool path planning algorithm is stored in this folder which can only be accessed only through administrative rights. While the information regarding the positioning and aesthetics of the web page is stored in *Default.aspx*, which can be accessed by users.

The client web browser will send an information with an activation command, and the

CAD server will validate the client command and talk with database and CAM system. The CAD server will pass a "query" to initiate CAM macros for generating tool path for manufacturing the sign. The *query* contains an information regarding the location of CAD model and STL file in a database and the information regarding option chosen by the user. If the *Make sign* option is passed in a query, the CAM macros will generate the tool path using standard tooling and step-size, and save the tool path in the same directory in the database from where the STL file was picked. Similarly, If the "Make tool path" is chosen, then CAD server will redirect client to another webpage. Here, some information regarding client's CNC machine, tooling required and output format of G codes is taken from the user, and the custom tool path is generated for that specific user. This tool path is stored in same directory, which is later sent to the user.

Once the tool path is generated and stored in a directory, the CAM macros will send the link for downloading G codes file to the user by email or to the CNC machine for manufacturing. The above section gives a complete description of the inter-connectivity between CAD server and CAM server with the client web browser. The CAM macros were implemented using the method described above and further optimization of tool path and optimal cutter selection was conducted to reduce machining time and improve quality which is discussed on chapter 4.

Chapter 4

Tool Path Optimization

The traditional CNC machining methodology generates tool path trajectory based on the user defined cutter location points that depends upon the type of surface being machined. The problems related to machining time, high scallops are observed in these methodologies with a fixed path which is discussed in Chapter 2. Moreover, these methodology do not have in-built gouge checking. In the improved methodology described in Chapter 3, the tool path planning is separated into two steps: tool path foot print; and cutter positioning. By separating the tool path trajectory planning into foot print description and cutter positioning, the tool path planning has an inbuilt capability of gouge checking and an ability to deal with interfering features. The foot print to machine a feature, namely a complex surface with a design feature can be identified at design stage. During the design phase, the geometry of the shape is known and hence planning the path of the tool (tool path foot print) is possible and can be programmed with creation of the feature. The final design may require multiple features, some of them can interfere with one another. In the traditional CNC machining methodology the interference among the features would lead to gouging of the surface. The separation of foot print and cutter positioning allows the features to interfere without gouging the surface.

This method allows surface to be machined with desired accuracy independent of type of cutter used. For a wooden sign of 432mm x 153mm x 25mm machined using a ball nose end milling cutter as described in Chapter 2, the machining time was found out to be 150 minutes and the average scallop height was found to be 0.01657 mm. The total number of tool passes required machining this sign was 384 and the tool path length was found out to be $10,611,916.8$ mm. Machining the same surface with improved methodology with flat-end mill and radiused end mills, could result in lower machining time. The same sign with flat background having majority of flat portion can be machined with the flat-end mill having the same radius and side-step resulted in scallop height of 0.01162 mm. The same sign was machine with radiused end mills having same radius and side-step and the scallop height was found to be 0.01067 mm at same tool path length. Since the scallop height is much lower, the side-step for machining a surface with radiused end mills and flat end mills can be increased. This will reduce the total number of passes and reduce tool path length required to machine a sign. It can be seen that for machining a part whose majority of the surfaces are curved, the ball nose end mills produces better results. In reality the machining surface is composed of both flat and curved portion in varying proportion. Thus, it may be possible to reduce machining time and tool path length by using optimal set of cutter for machining the surface.

With increase in total number of tool passes, the total number of points in the tool path will increase. At each of these point the cutter is dropped in simulation and intersection calculation have to be performed. This will increase the computational time which will ultimately increase tool path generation time. Thus, optimization of tool path is necessary to achieve lower machining time and lower tool path generation time.

The tool path optimization is carried out in two stages: optimization of tool path generation time; and optimization of machining time. The optimization of tool path generation time is carried out using “Bucketing Technique” which isolates the triangles to reduce surface intersections and reduce tool path computational time. The optimization of machining time is performed by implementing “Optimal Cutter Selection” strategy based on scallop height calculation and volume removal calculation. By selecting optimal set of cutter for machining sculptured surface, the final surface finish and machining time can be improved. By optimizing tool paths, the machining process for sculptured surfaces the method can be improved and can be made commercially viable.

4.1 Optimization of tool path generation time

A tool path typically comprises of many line segments, each of which represents one linear movement of the cutter. The sculptured-surface machining typically involves more than 10,000 tool movements, for a typical surface with the number of points ranging between 3,000 and 50,000 points. At each of these 50,000 cutter positions the dropping algorithm has to be executed and the whole process can be lengthy. Considering this, a method was developed to optimize the tool path and process only those triangle which have a higher possibility of intersecting the cutter at any particular position. This technique of isolating the triangles which has possibility of intersection with the surface at any particular cutter location to reduce tool path generation time is called Bucketing Technique.

For bucketing a surface, a point consisting of x, y and z coordinates, and the area of region is defined. The entire surface of the work piece is divided into separate areas (regions). For each regions, the number of triangles that lies inside the region is found. At each tool position, the tool axis is localized within the region in which cutter lies. For that cutter position, only the triangles in the region in which cutter lies is checked. As each subsequent tool movement is processed, the cutter is localized in a particular region based on its location over the work piece surface.

Since a given tool movement will intersect only a small percentage of the triangles, it is highly desirable to eliminate from consideration all those triangles which has no possibility of intersection with the cutter. A regular 3-D grid of rectangular cuboids work piece in x-y-z space is used. Since all the triangles with normal vectors are chosen such that they point towards the positive z direction, so the rectangular regions are drawn only in x-y plane. These small rectangular cuboids represent a region which is considered as "buckets" containing triangles representing work piece. These buckets are shown in Figure 4.1. If the feature is complex, the triangle vertices are dense and each bucket will contain large number of triangles in bucket. Conversely for a simple feature the vertices of the triangles are sparse, each bucket will contain few (but greater than zero) triangles in that particular bucket. All the triangles falling in a given bucket will be stored in a linked list corresponding to that bucket. A given tool path, passes over a certain set of rectangular bucket, and all points lying within these buckets must be examined. This set of buckets is easy to determine because the rectangle boundaries are a regular $x - y$ grid.

4.1.1 Bucketing Optimization Algorithm

This section describes how the work piece triangulated surface is subdivided into different regions to generate gouge-free tool path with minimum processing time. The tool path generation time is directly proportional to the work piece size and shape, cutter size, type of cutter and step-over distance. With increase in complexity of the work piece surface the number of triangles and their intersection calculations would grow, which would ultimately

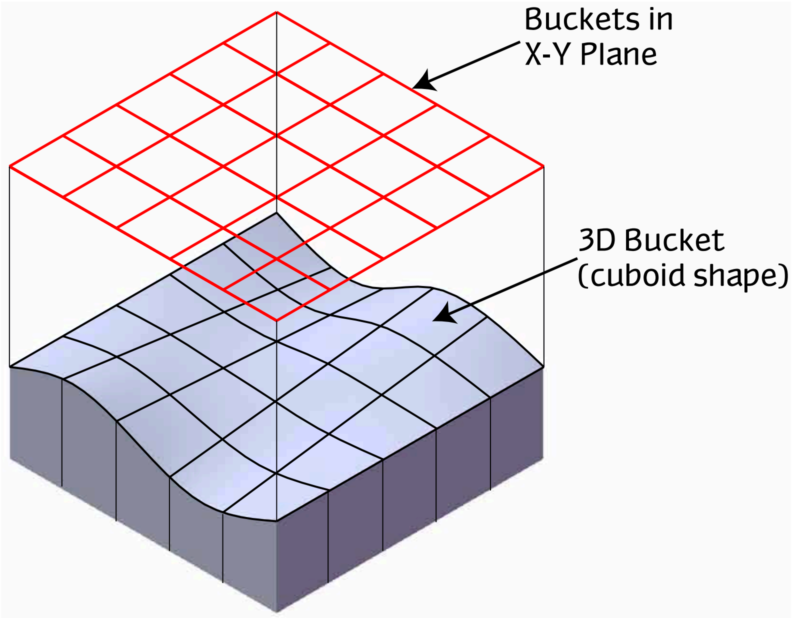


Figure 4.1: Work piece Surface Subdivision into small buckets

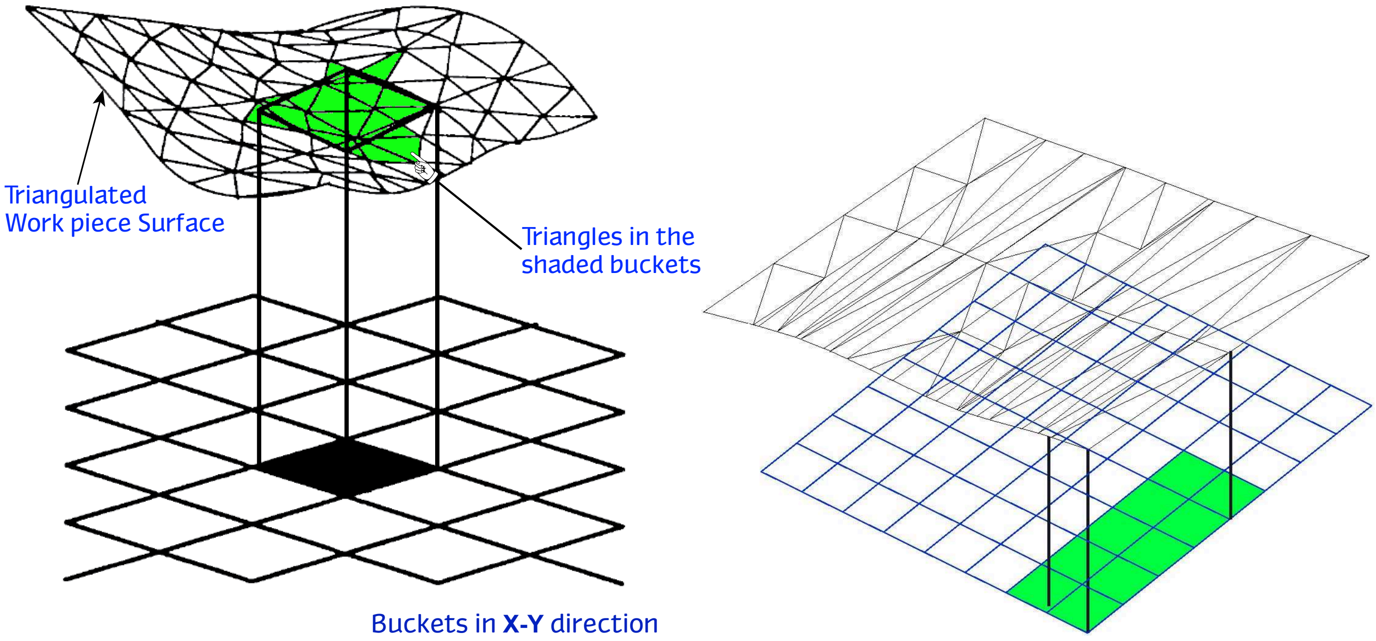
increase tool path generation and machining time. Therefore, the CPU load grows linearly with the number of movements based on cutter size and step-over distance.

There are three key aspects in the bucketing algorithm, *Discretization*, *Localization* and *Intersection*. In the *Discretization* phase, the surface is divided into small buckets in such a way that entire work piece is covered. The size of the bucket depends upon the size and shape of the cutting tool, desired accuracy and the local surface curvature. Each bucket is of same size which causes uneven number of triangles in each bucket. Once the size and number of buckets are decided, each bucketed region is stored with a unique *Bucket Number* representing its position relative to origin of the work piece. The surface normal at the centre point of each bucket determines the bucket's view direction. The discretization phase is shown in Figure 4.1. The $buck_x$ and $buck_y$ represents the number of buckets along X and Y axis respectively.

```
for $i = 1$ to $buck_x$ do
  for $j = 1$ to $buck_y$ do
    Represent each Bucket with a unique $Bucket\ Number$
  end for
end for
```

During the *Localization* phase, the total number of triangles of the work piece in which each bucket lies is calculated. It is computationally time-consuming to check for possible intersection of cutter with all the triangles for each cutter position. Thus, to eliminate the triangles which have no possibility of intersecting the cutter while machining, the localization phase is performed. The localization phase lowers the total number of triangles to be checked for intersection at each cutter position. Each triangle is checked to see whether it lies in the bounds of the bucket. All the buckets in which that triangle lies are recorded in an array, and this process is repeated for all the triangles. Some of the triangles are big, and will lie in more than one bucket. An example of both the cases where a bucket contains many triangles and where a triangle is in more than one bucket are shown in Figure 4.2a and Figure 4.2b respectively. Later, the huge data is sorted in such a way that all the triangles are separated with respect to bucket number in which it lies. The diagrammatic representation of all the triangle lying in buckets is shown in Figure 4.2. The pseudo code for finding all the triangle in a particular bucket is shown below.

```
for $i = 1$ to *no of triangles* do
  $bounds1 = BucketNumber$ in which $\overline{V}_1$ lies
  $bounds2 = BucketNumber$ in which $\overline{V}_2$ lies
  $bounds3 = BucketNumber$ in which $\overline{V}_3$ lies
```



(a) Triangles inside the Bucket

(b) A triangle inside more than one Buckets

Figure 4.2: Representation of triangles inside the Bucket

```
        $buck_{min}$= minimum of $bounds1$, $bounds2$ and $bounds3$
        $buck_{max}$= minimum of $bounds1$, $bounds2$ and $bounds3$
        **for** $j = buck_{min}$ to $buck_{max}$ **do**
            Add Bucket Number
        **end for**
    **end for**
    Find All triangles in bucket
    Store in an array
```

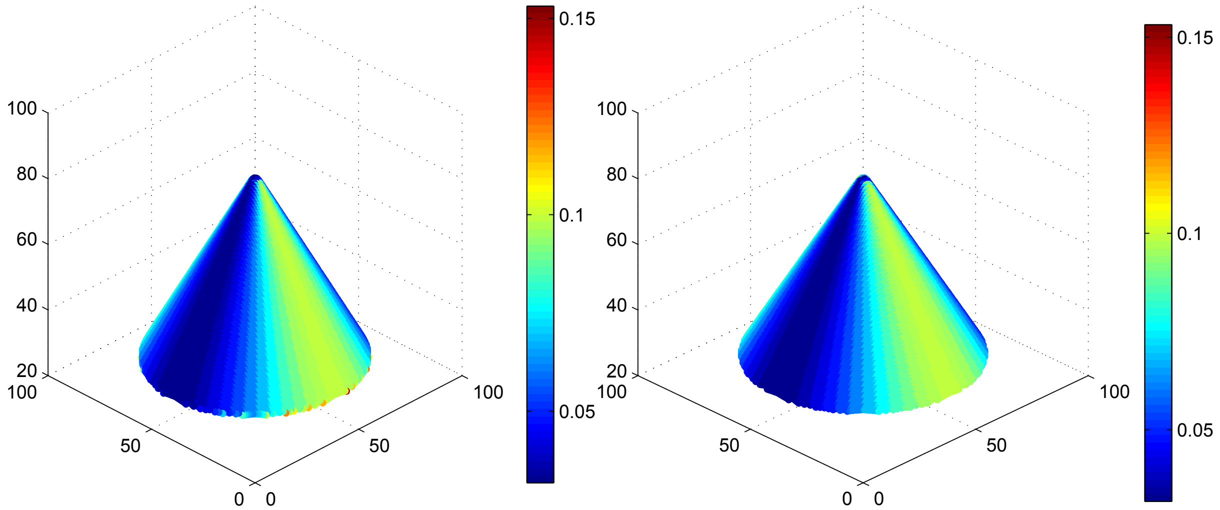
For each cutter position a check is made to identify which buckets lie in its shadow. The schematic diagram of the projection of cutting tool onto buckets is shown in Figure 4.2. In Figure 4.2 it can be seen that a cutting tool lies in the buckets represented by black shaded area. Localization is achieved be finding the set of buckets which lie under the shadow of the end milling cutter and examining triangles in those buckets only. Thus, only a small number of triangles are examined for each cutter position.

In the *Intersection* phase the cutter is intersected with the bucketed triangles and the gouge-free cutter location is calculated. The procedure for intersection is explained in Section 3.4. The pseudo code for checking the triangles in the buckets is shown below. The total number of triangles after applying bucketing algorithm (*ntriangles*) for particular cutter position is 75% less as compared to total number of triangles in an STL file.

```
Tool Centre $T_C$ with coordinates $TC_x$, $TC_y\ and\ TC_z$
tempbuck= Bucket Numbers in which $\overline{T_C}$ lies
ntriangles= Triangle numbers in tempbuck
use ntriangles for all checks
```

For analyzing the result obtained, the tool path was generated using the method described in Chapter 3, using a zig-zag tool path footprint. The tool path was generated before and after applying bucketing algorithm, to measure the effectiveness of the bucketing algorithm. The results obtained were within the specified tolerance limits of the "Dropping Method" without applying bucketing algorithm. This subdivision method requires fewer points for a given level of accuracy, resulting in about 75-80 % of the CPU time required to generate tool path. Another critical analysis was conducted to check the improvement in surface finish obtained by using Bucketing Algorithm. The variable side-step is employed instead of same side-step based on the bucketing and scallop height data to improve the final surface finish of the product. The scallop height of the final machined products after applying bucketing was lower compared to the actual scallop heights.

In this analysis, the tool path was generated with each bucket being of the same size and fixed dimensions. The scallop height analysis is conducted with the tool path generated with the fixed bucket sizes. The total scallops height in each bucket is calculated and the buckets are sorted from highest scallops to lowest. All the buckets which have scallops more than average scallops of the work piece are chosen, and for those areas the tool path is generated with a reduced side-step. This will lower the total scallop height of the work piece. Thus, by combining the results of bucketing and scallop heights, an efficient tool path can be generated with best surface finish.



(a) Scallop Height with same side-step

(b) Scallop Height with varying side-step

Figure 4.3: Scallop Height after applying variable side-step

To demonstrate this analysis a tool path was generated with a ball nose end milling cutter with Radius of $1/8^{th}$ inch and the side-step of $1/40^{th}$ inch with 25 buckets each of 20 x 20 mm. For the buckets which resulted in higher scallops, the tool path was generated with same radius and number of buckets but the sidestep was $1/64^{th}$ of an inch. The results of the ball nose cutter with constant side-step, and with lower side-step applied to the buckets having higher scallops is shown in 4.3. The total scallop height of the work piece was reduced from 759.954 mm to 740.05 mm, and the maximum scallop height was reduced from 0.153 mm (6.027 thou) to 0.0984 mm (3.874 thou). There is about 5 % of reduction in total scallop height by performing this analysis. Moreover, it was observed that the buckets having higher scallops have large variation in the surface depth i.e. variation of Z axis levels and the bucket with lower scallops have comparatively less variation in depth. Thus, it can be deduced that the areas where the depth variation is large, the scallops height will be higher, so these areas should be re-machined or they should be machined with a lower side-step. By employing this technique, the component with a better surface finish can be generated.

4.2 Optimization of machining time

After a part is designed, a sequence of operations (process plan) needs to be determined to transform the raw material into the final product. For generating tool path using the methodology discussed in Chapter 3, the foot print is developed at the design stage and the gouge-free cutter location is found at the points along that foot print. For machining any work piece using this methodology, the foot print along which cutter moves remains the same. Thus, either ball nose, flat-end or radiused end mills can be moved over that foot print. It would be beneficial to choose a cutter to move over the foot print which results in best surface finish and machining time. Thus, Cutter selection is one of the key tasks in process planning, which affects the productivity and final surface finish.

Most of the available commercial CAM packages require the user to select appropriate cutters, but it is nearly impossible for a user to determine what may constitute an optimal cutter combination for a given surface without interference checking and surface finish assessment. Vickers and Quan [6] compared the ball-nosed and flat-end milling cutters for machining of curved surfaces, but the machining time was very high. Mizugaki *et al.* [23] reduces the machining time, but the final surface finish produced is not within acceptable tolerances for sculptured surfaces. Furthermore, the verification process is expensive and time-consuming. To avoid potential problems associated with gouging and collision, the user is often forced to make a conservative choice that results in higher machining time and high production cost. Thus, the tool path must be generated which produces the surface within tolerances with minimum machining time.

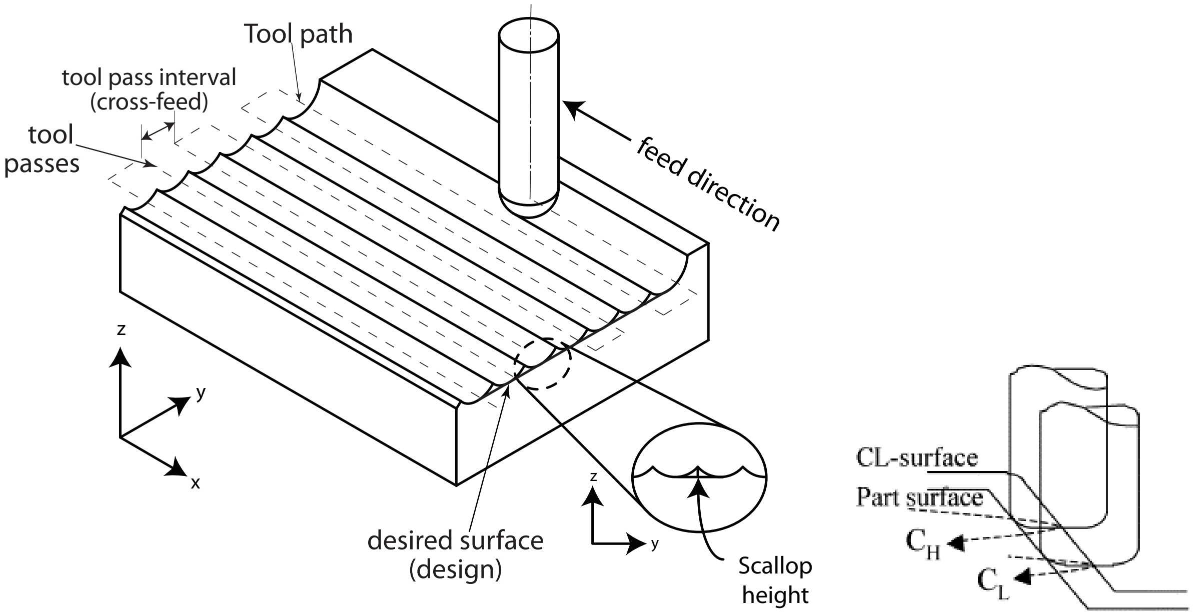
When the cutter is moved along the foot print, there is some amount of material left-over between two consecutive cutter positions. The surface should be machined with minimum material left-over. Scallop height is one of the major problem observed in "Ball-Drop" method developed by Manos *et al.* [14]. The machined sign using "Ball-Drop" method resulted in very high scallop heights which is discussed in 2.1.3. Thus, it will be beneficial to compare the tool path generated by different cutters based on scallop height. The detailed discussion of calculating scallop height for different cutter is discussed in Section 4.2.1.

The cutter positioning is performed using "Dropping Method" is based on best cutter depth i.e. it is dependent upon the lowest the cutter can go without gouging the work piece at any particular position. Thus, the cutter position chosen by the tool path represents minimum volume of material left on the work piece. For checking volume of material removed by ball nose, radiused and flat end mills, a surface developed by a tool path that has minimum volume left-over after machining is used. Thus, it is a good way for estimating volume of material removed after machining. The cutter volume removal algorithm is discussed in Section 4.2.2.

4.2.1 Scallop Height Algorithm

When the cutter moves along a tool path, any material that falls within the cutter envelope surface is removed. For the ball-end mill, the envelope can be considered as a cylinder which is the envelope of the spheres of radius R centered at the CL path. The cylindrical surface is generated by a set of circles, of radius R, which are the intersections between the spheres and the planes perpendicular to the directions of motion of the tools centres. The *scallop curve* is generated as the intersection curve between two adjacent cutter envelope surfaces. The highest point on the scallop curve relative to the design surface is called a *scallop point*. Since the scallop curve is defined on a cutter envelope surface, a tangent vector of the scallop curve at a scallop point is also a tangent vector of the cutter envelope surface at the scallop point. Moreover, it is also a tangent vector of the sphere which contains the scallop point. The distance between the design surface and the scallop point represents *scallop height* (h). The detailed representation of scallop height and other important parameters are shown in Figure 4.4.

Given that the feed direction of the tool path is along the $x - axis$ the scallop heights are measured on the $yz - plane$. The ball nose cutter is modeled as a two dimensional circle with radius R. The inserts in the radiused end mill are also modeled as two dimensional circles with radius R_1. In order to determine the scallop height a tangent line connecting two successive tool positions along the $yz - plane$ is found. Let the center point of the two tool positions modeled as circles are (y_0, z_0) and (y_1, z_1) with radius R. The endpoints of



(a) Scallop Height

(b) surfaces in scallop cutter

Figure 4.4: Scallop Height in a machining face

the common tangent to both these circles are given by:

$$\begin{aligned} y_{st} &= y_0 + (R\cos\alpha) \\ z_{st} &= z_0 + (R\sin\alpha) \\ y_{end} &= y_1 + (R\cos\alpha) \\ z_{end} &= z_1 + (R\sin\alpha) \end{aligned} \tag{4.1}$$

Where angle α can be found from the Equation 4.2:

$$\cos\alpha = \frac{(y_1 - y_0)R^2 + (z_1 - z_0)\sqrt{((y_1 - y_0)^2 + (z_1 - z_0)^2 - R^2)}}{(y_1 - y0)^2 + (z_1 - z_0)^2}$$

$$\sin\alpha = \frac{(z_1 - z_0)R^2 + (y_1 - y_0)\sqrt{((y_1 - y_0)^2 + (z_1 - z_0)^2 - R^2)}}{(y_1 - y0)^2 + (z_1 - z_0)^2}$$

Once the end points of the tangent are found, lines are drawn in the z direction between these two points.

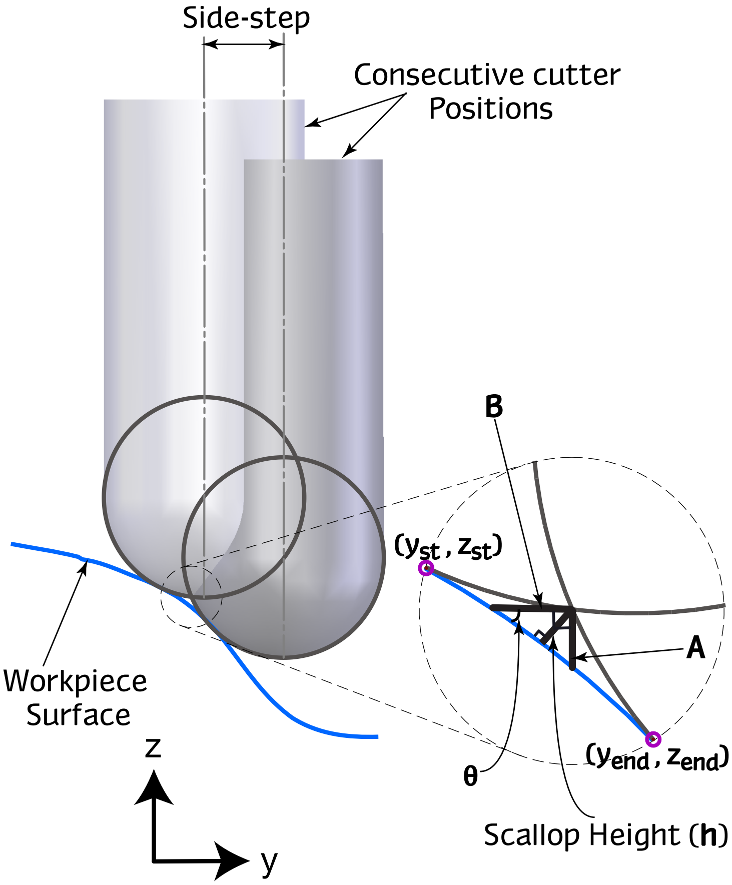


Figure 4.5: Scallop Height Calculation for two consecutive cutter positions

The coordinates of Y-axis is varied from y_{st} to y_{end}. In the case of ball nose, the intersection of each line representing work piece surface is found with the circles representing the cutter at the given positions. The intersection is performed at each of these regions, and the intersection points are calculated. Once the intersections are calculated, the point with the minimum z value is chosen as the desired intersection point; this ensures that the intersection point of the line and the circle is at a valid position. The distance, along the Z-axis, between the intersection point and the work piece surface represents distance A. Also, distance B is defined by the line along the y-direction connecting the intersection point and the work piece surface. Based on the values of A and B, angle θ and scallop height h are calculated. The diagrammatic representation of scallop height calculation for ball nose end mills is shown in Figure 4.5. The maximum value is chosen as the scallop height for two successive tool positions. This procedure is than repeated for all other tool positions. This scallop detection algorithm was implemented for the zig-zag tool path foot print.

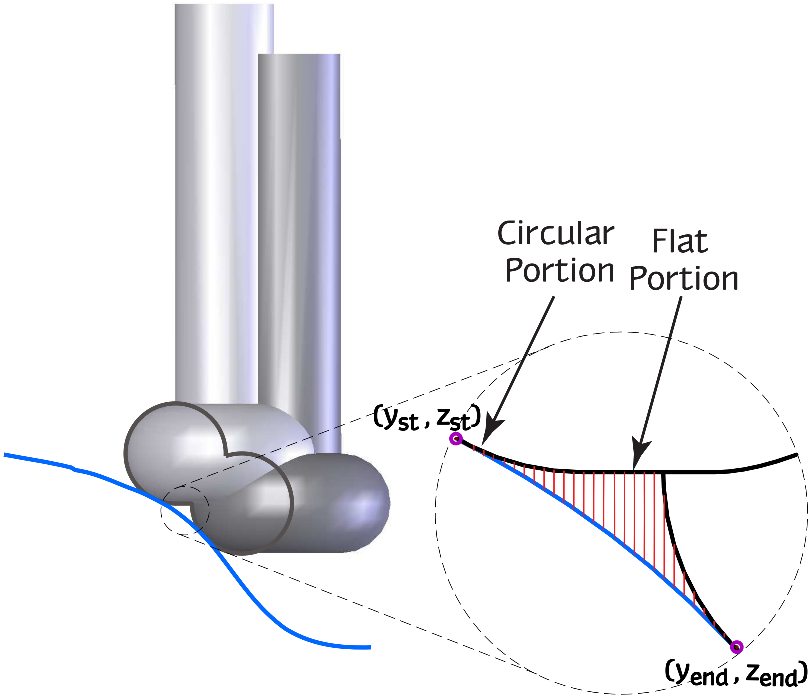


Figure 4.6: Scallop Height calculation in Radiused end milling cutter

In case of the radiused end mill the intersection point of each line shown in red color in Figure 4.6 is found individually for both inserts, represented by circles, and the flat portion connecting the inserts. The procedure for calculating scallop height for radiused end mills are similar to that of ball nosed cutters except the geometry of the intersection of two cutters is different. As shown in Figure 4.6, the intersection geometry contains circular region and a flat region. Thus, a simple *Circle-Circle Intersection* method could not be applied. For the cases where the two consecutive scallops heights are at same z-value then the *Circle-Circle Intersection* could be applied depending upon the centre-centre distance between two cutter position. If the centre-centre distance is between 0 and $2R_2$, there will be surface contact between cutter and work piece i.e. the flat portion of radiused cutter would intersect flat portion of workpiece. This condition would result in *zero scallop height* so no material is left between this cuter position. Similarly for the cases where the centre-centre distance is lies between $2R_2$ and $2(R_1 + R_2)$, the scallop height can be calculated using circle-circle intersection method modeling radius R_1 as two-dimensional circle. This would result in scallop height same as that of ball nosed cutters. But when the two consecutive cutter are located at different z-levels then a new strategy is employed to calculated scallop height which is discussed in next paragraph.

In order to calculate scallop height for the radiused cutter, lines are drawn at specific interval in the direction parallel to the tool axis i.e. z-axis. Each vector corresponds to a blade of grass growing from the desired object. These set of directional vectors are shown in red color in Figure 4.6. The coordinates of the lines along z-axis are varied along the y direction from y_{st} to y_{end} by predefined steps. The values of y_{st} and y_{end} can be calculated from Equation 4.2. Mostly, the incremental step of the vectors is $80^{th} - 100^{th}$ that of the side-step with which the tool path was generated based on the accuracy of result required. For each vector, a line is extended from the tangent line between two surfaces and the first contact point with the cutting tool geometry is calculated. Later, the procedure is similar to that of ball nosed end mills. The value of A, B and θ are calculated, and from these values scallop height h for the radiused end mills is found. This procedure is repeated for all the vectors from y_{st} to y_{end} and the maximum of all scallop values, is chosen to be final scallop height for radiused end mills.

Scallop Height is a good way to determine which tool is better at a given area. Material left behind in one tool position can be removed at another tool position. For a global perspective a second factor should be used to evaluate the surface quality. This factor is the amount of cutter volume removed resulting from the machining process.

4.2.2 Volume Removal

When machining a surface it is desired that the amount of stock material leftover in the finished product be minimal. A comparison of volume removed by each cutter is calculated in order to determine which cutter is best suited for machining a specific surface point. At any given position on the surface being machined the tool position of a radiused cutter and a ball nose cutter can be found using the methods described previously. Using the "Dropping method" for ball nose and radiused end mills, the cutter location is found at the lowest point possible without gouging the surface being machined. This means that the cutter position chosen represents minimum volume left over by each tool without gouging the part. The method that will be described in this section determines which tool, ball nose or radiused, removes larger volume at a given x, y position.

Here, the tool path is based on drop the ball cutter location method i.e. it depends upon the lowest the cutter can take without gouging the work piece at any particular tool position. In other words the cutter position represents the volume that will be left on the work piece. The algorithm for tool volume removal for the tool path developed using "Dropping Method" for ball nose, radiused and flat-end mills is developed. The minimum volume left-over after machining is a good way to compare the volume removed by ball nose, radiused and flat-end milling cutters.

Here the two tool positions are superimposed in Figure 4.7. The ball nosed cutter is shown to the left and the radiused end mill is shown to the right. The distance between the center of the *ball* and the *doughnut* is designated as d. The value of d_{base} is found at the point at which the area removal for both tools is the same, i.e. the ball area equals the torus area. The objective is to maximize the material removal. But, since tool is a 3 dimensional object, the volumetric material removal is a better indication of the quality of machining. The detailed explanation and Equations are shown in Patel *et al.* [9].

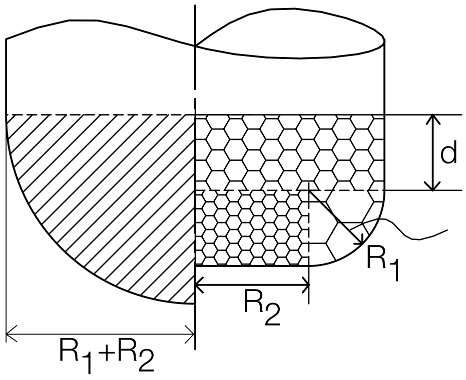


Figure 4.7: Volume comparison for ball nose and radiused end milling cutter [9]

In conclusion, if the d at a cutter location is greater than the d_{base} then the radiused cutter removes more material and is preferred. Otherwise, the ball nose cutter has greater material removal and is preferred.

4.3 Simulation Results

The simulation is a process whereby a machine model with a graphical interface is created so that it behaves much like an actual CNC milling machine to verify the effectiveness of the manufacturing process and the tool path. By simulating the machining process offline literally thousands of dollars are saved in down time, scrapped parts, broken tools and machine crashes while improving overall machine productivity. CNC simulations are widely used by manufacturers and job shops to keep their CNC machines running reliably and with minimum errors. For this research, the tool path was verified and machine simulations were carried out using a custom machining simulator ***ToolSim*** developed by Israeli [5]. ToolSim was used to simulate the machining process, and simulation error was found to be within limits as specified in Mann *et al.* [19]. For the Toolsim, confirmation experiments were conducted by Israeli [5] which showed that the results of the simulation generally match the results of the real machine, within the tolerance of the digital laser scanner.

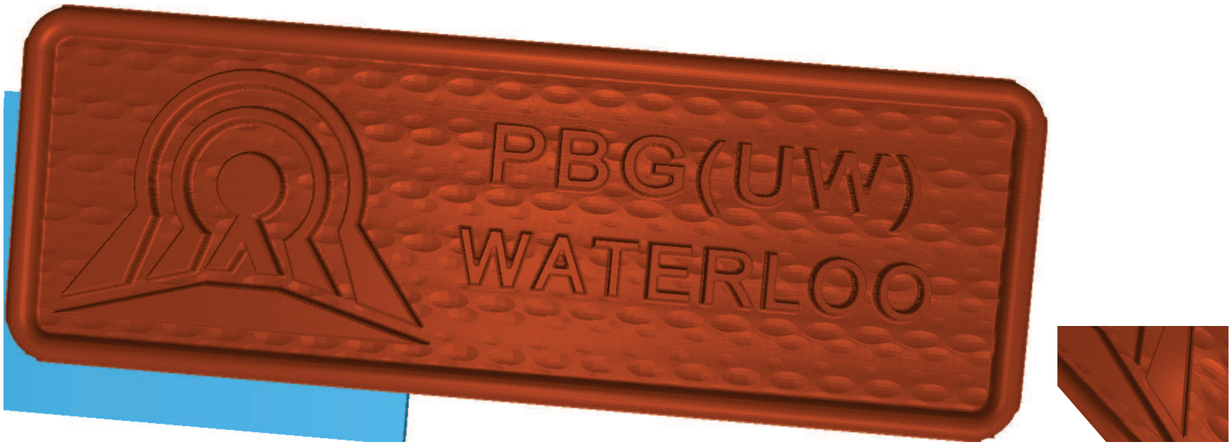


Figure 4.8: Simulation Results on ToolSim using ball nose end mill

The simulator was developed by computing the surface swept by the cutter i.e. by determining curves on the cutter whose imprint will be left on the stock as the cutter moves. To demonstrate these ideas, tool motion code is integrated with the in-house NC machining simulator and the simulated results are compared to mathematically representation of the surface for a tilt/rotary five-axis machine. The computational geometric model of the work piece is cut away by cutter swept volumes for 3D simulation or cutter swept areas for 2D simulation. To compute the mathematically exact tool motion, the matrices representing the moving parts of the machine are linearly parameterized. The composite transformation is non-linear and complex for computing the cutter motion, the computation of this motion is decomposed by traversing the tree and computing the motion during this traversal. Such an approach would allow to essentially compute the motion as the composition of a series of linear matrices. The detailed discussion regarding the procedure and accuracy of the simulator is shown in [19].

A tool path is checked in ToolSim before actual part is machined. A simulation of the PBG sign machined with ball nosed end milling cutter is shown in Figure 4.8. In order to verify the effectiveness of the tool path in the simulator regarding cross cut and other nearest triangle surface error estimates were performed. The simulation results obtained from the ToolSim were closely match to the actual CAD model which is shown in 4.8.

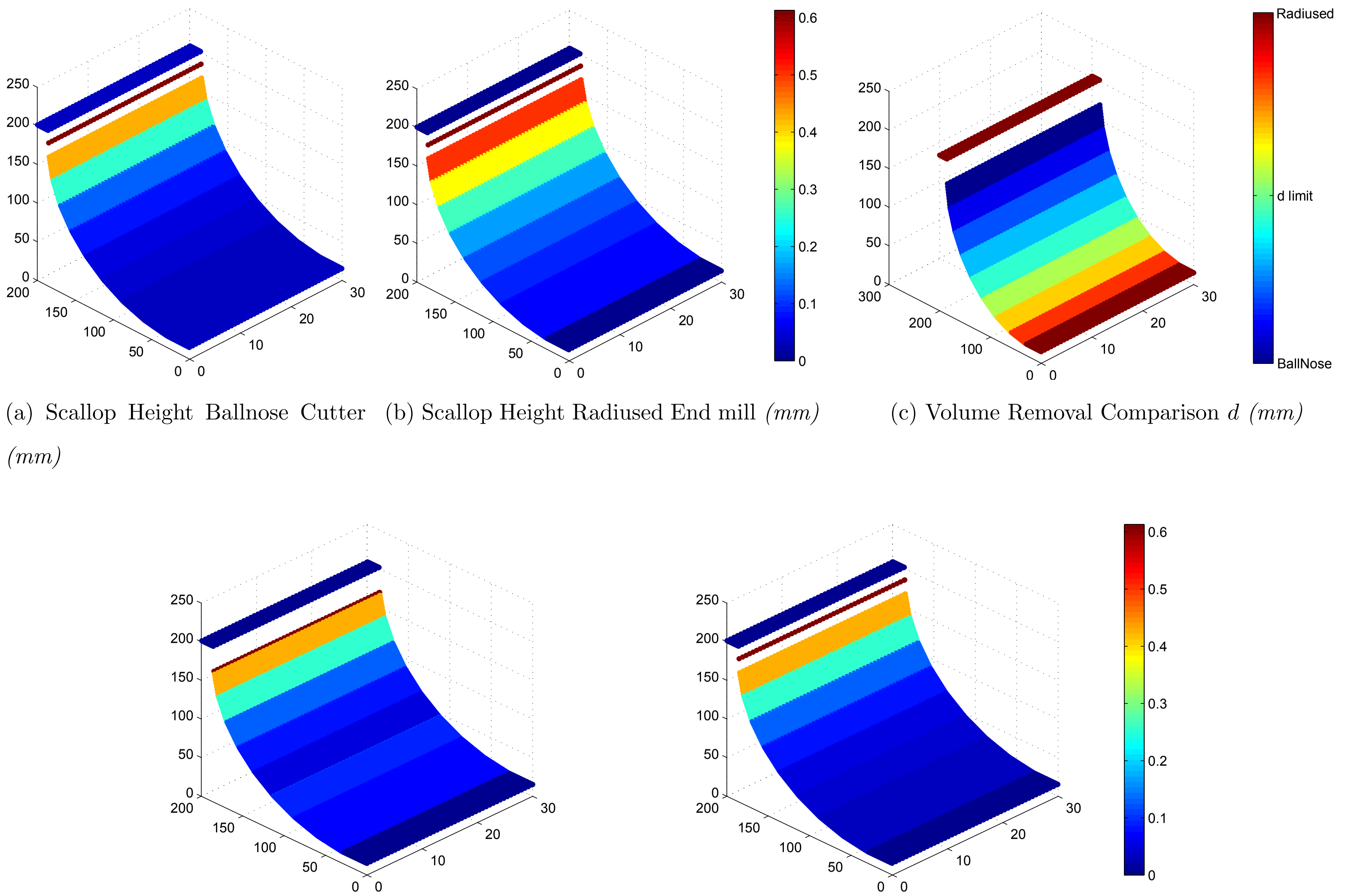
4.4 Confirmation Results

To evaluate the performance of each type of cutter a sign was machined in three stages. The surfaces, shown in Table 4.1, were machined using both tools. Scallop height and volume removal are calculated for each of these surfaces using the procedures described in the previous section.

For understanding the relationship between the angle of inclination with the type of cutter resulting in best surface finish, a surface with varying angles from 0°-90° in steps of 10 was machined. The part was machined with a $1/8^{th}$ inch ball nose end mill and $1/8^{th}$ inch radiused end mills with major and minor diameter of $1/16^{th}$ of an inch. Figure 4.9 shows the comparison of ball nose and radiused end milling cutter for better surface finish wrt angle of inclination to the tool axis. It shows how the portions of the spherical and toroidal part surfaces making an angle more than 36° with the tool axis are best machined with a ball nose cutter and similarly those portions that make an angle less than 36° are best machined with a radiused end mill. The transition occurs when the surface normal makes an angle of 36° with the tool axis. In order to achieve exact value of transition point another test was conducted with a surface designed with angles varying from 35°-40°.

The study was further conducted for same part but with different radii of ball nose and radiused end mills and the transition was found in the range of 35°-40°. The radius of ball nose and radiused cutter was varied to $1/32^{nd}$ inch and $1/8^{th}$ inch, and the transition point was shifted a bit from 36° to 38°, but the transition range was still the same. Thus it can be deduced that with increase in angle the transition increases, but the variation is very small.

Figure 4.9a and Figure 4.9b show the scallop heights for the ball nose and radiused end milling cutter respectively. The volume removal comparison for each surface is shown in Figure 4.9c. This is obtained by comparing the value of d (volume removal) for both tools at a tool position and selecting the one with the highest material removal. Values higher than d_{base} have higher material removal when machined with the radiused cutter and vice versa. Figure 4.9d shows the scallop heights resulting from machining the surface based on the results of the volume comparison. Both cutters are used to machine the surface, for each cutter location the cutter with the highest volume removal is selected. The result is a machined part that will have the least material leftover. The tool movement is not considered. Finally, Figure 4.9e shows the surface being machined but in this case the tool that results in the smallest scallop height is selected. The result is a machined part with minimal scallop heights. Scallops depend on the direction of tool movement. This method assumes that the tool path footprint is known and cannot be changed.



(a) Scallop Height Ballnose Cutter *(mm)*

(b) Scallop Height Radiused End mill *(mm)*

(c) Volume Removal Comparison d *(mm)*

(d) Scallop Height Resulting from Maximum Volume Removal

(e) Scallop Height Resulting from Minimum Scallop Height

Figure 4.9: Scallop height and volume removal variation with increasing angle between the z-component of normal vector of surface and the tool axis

The relationship among the angle between the tool axis and the surface normal is a useful result for machining sculptured surfaces efficiently. Since the spline surfaces are mostly composed of areas where the angle between the surface normal and the tool axis is large, the ball nose was determined to be better suited for these surfaces. It was found that as the angle increases scallop heights increase for both types of tools. For the ball nose cutter the scallop heights do not increase significantly until the angle is greater than 60° as shown in Figure 4.9. In the case of the radiused cutter the scallop height increases as soon as the angle is greater than zero. It then remains constant until the angle is greater than 40°; at this point scallop heights increase drastically. Similar results were observed in the volume removal calculation. In this case it was found that at 36° the ball nose tool starts removing more material then the radiused cutter. The transition in volume removal from a radiused cutter to a ball nose cutter can be explained by that fact that at 36° the removal volume on the radiused end mill is at most $\frac{1}{4}$ of a torus geometry defining the cutter, while the volume of removal for the ball nose end mill is half a sphere as was explained in section 4.2.2 of this thesis.

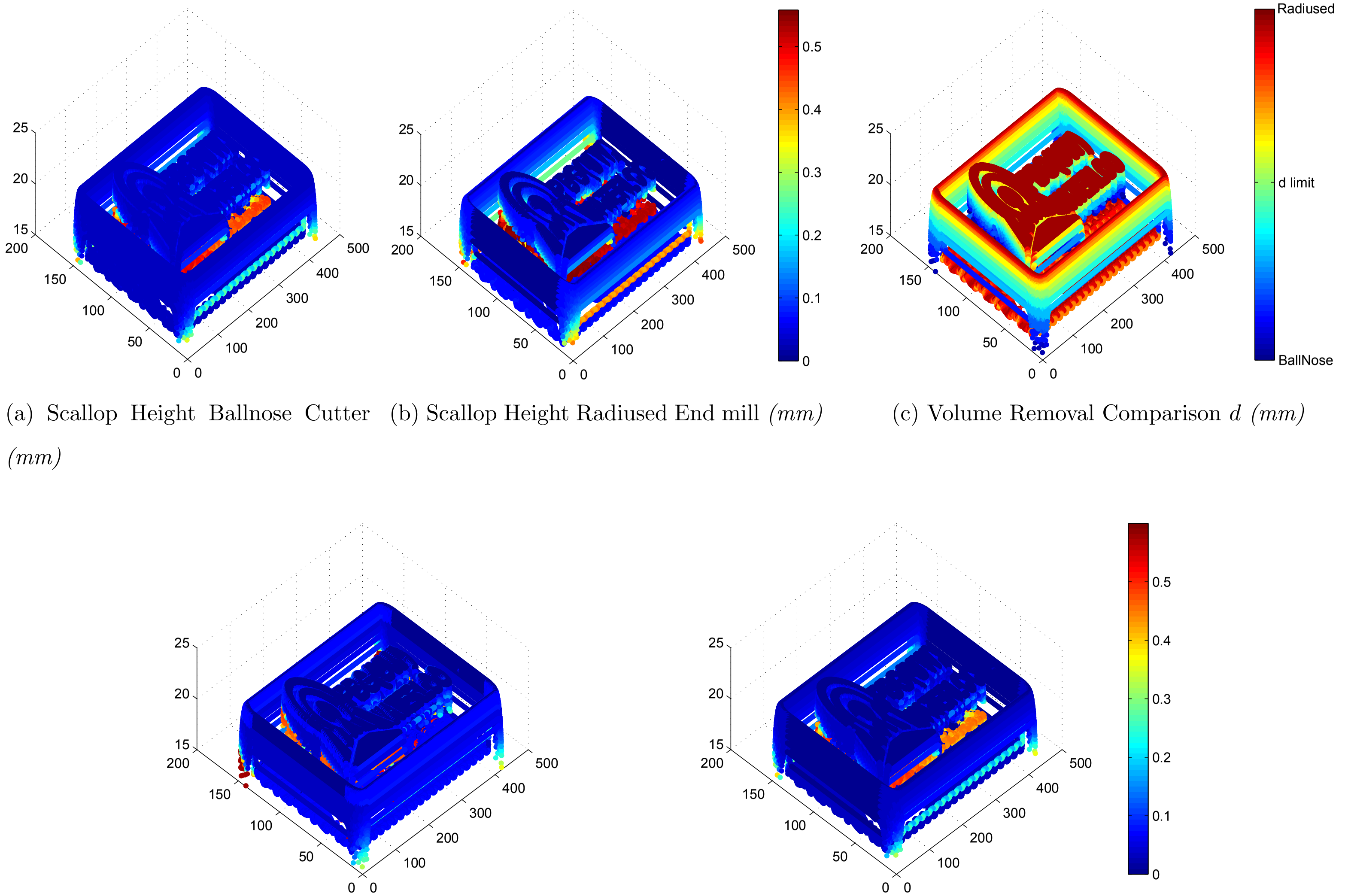
The variation in scallop height can be explained as follows. As the angle between the surface normal and the tool axis increases the motion along the z-axis also increases. In other words, when a surface is flat the side step only causes motion along the y-axis; however, when the angle increases the side step causes motion in the y-axis and the z-axis. This means that the distance between the two centers will be bigger. In the case of the ball nose end mill the effect is not that significant since the contact radius is large in comparison with the contact radius of the radiused end mill. Only the radius of insert is considered at this point since for any angle greater than 0° only the insert is in contact with work piece.

Cutting tests

The machining was conducted on real life applications for checking the efficiency of the process and make it more productive. Tests were conducted on a sample wooden sign with scalloped background and twine border to verify the conclusions drawn from the results. Two different wooden signs were machined for measuring the measuring the effectiveness of the cutter. First, a sign with a logo, text, spherical border and a doughnut-cut background and secondly, a wooden sign with scalloped background and twine border were chosen for analysis. These signs are the best example of the sculptured surface having complex surface geometries. In the first stage the design only includes lettering/logo. For the second stage the sign includes the same lettering as well as a border. The final version includes lettering, a border and an intricate background. The objective of these tests was to examine the performance of the described methodology in an actual customizable product. A similar analysis, as the one performed for the four sections, was performed for the three stages. The results of the tests are shown in Table 4.1, 4.2 and Figure 4.10, 4.11 shows the scallop height distribution for the third stage. The tests were conducted in three stages.

For the first stage the results show that the radiused end mill is best suited for the operation. It has a lower average scallop height and higher volume removal for most portion of the sign. The only areas for which the ball nose is found to be better are except around the boundary of lettering. Nonetheless, the high scallops in these areas is is obtained because that area is inaccesible with a radiused end mill. The radiused end mill is best suited for this surface since it is mostly composed of flat regions.

For the second stage of this test, addition of a twine border, the radiused end mill still has a lower mean scallop height due the large area composed by flats. The scallops



(a) Scallop Height Ballnose Cutter *(mm)*

(b) Scallop Height Radiused End mill *(mm)*

(c) Volume Removal Comparison d *(mm)*

(d) Maximum Volume Removal

(e) Minimum Scallop Height

Figure 4.10: Scallop height and volume removal variation for PBG Sign with logo, text, border and background

Table 4.1: Scallop height for ball nose and radiused cutter for PBG Sign. Side step along and perpendicular to feed direction is $\frac{1}{40}$ inch (0.635 mm).

	Ball Nose End Mill $R = \frac{1}{16}in$				Radiused End Mill $R_1 = \frac{1}{32}in,\ R = \frac{1}{32}in$			
Type of Surface	Scallop Heights [mm]							
	Max	Min	μ	σ	Max	Min	μ	σ
Text	0.4992	0.021	0.0429	0.0512	0.5308	0	0.0169	0.0725
Border and Text	0.5015	0.021	0.0473	0.0606	0.5564	0	0.0703	0.0821
Background, Border and Text	0.5041	0.0318	0.0466	0.0612	0.5581	0	0.0711	0.0819

Text	Border and Text	Background, Border and Text

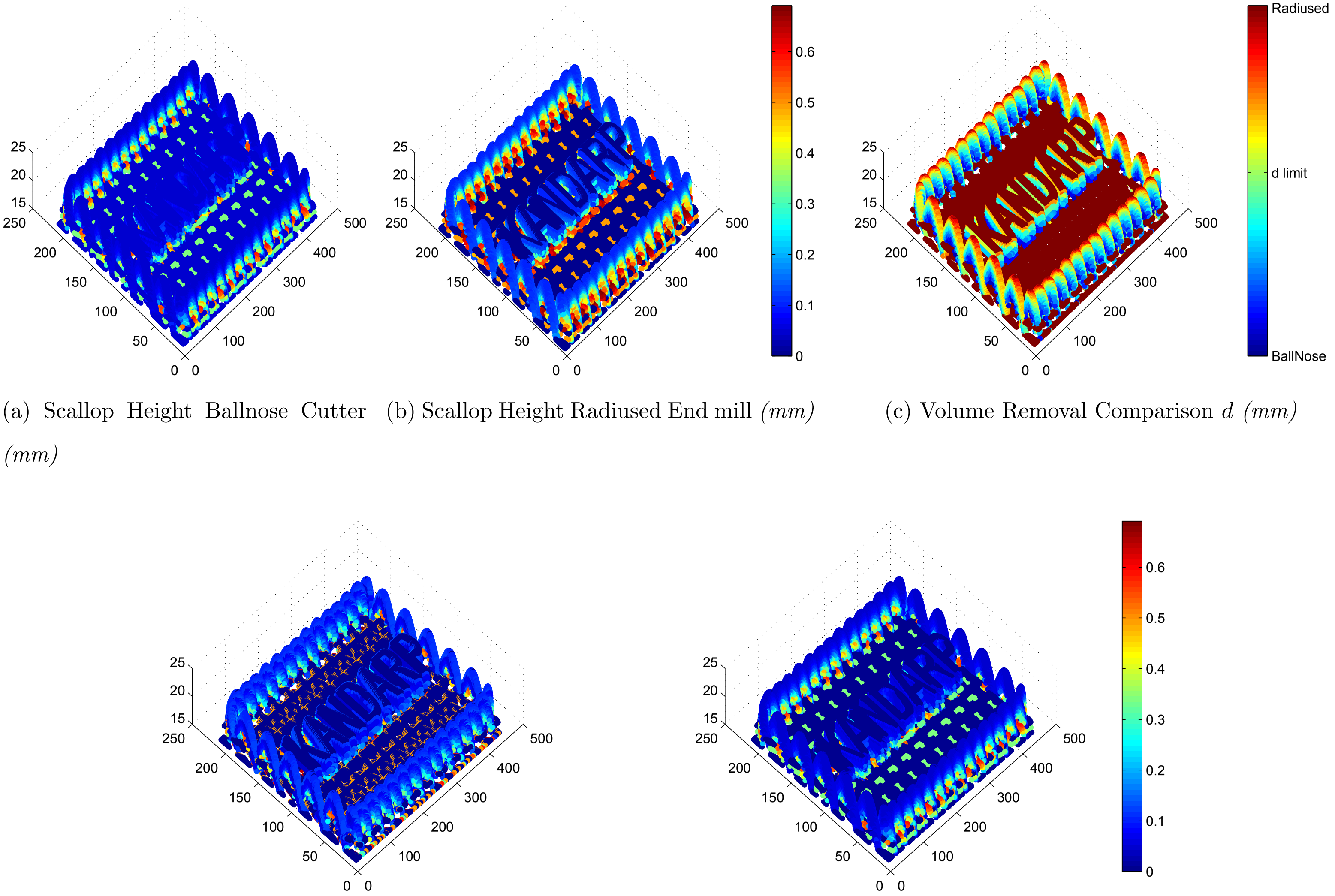
resulting from the machining of the twine are much larger for the radiused end mill as compared to ball nose. Furthermore, the volume removal is much larger for the ball nose end mill as the ball nose can cut deeper than the radiused end mill.

In the final stage of sign, on addition of a complex background, it was originally predicted that the ball nose cutter would provide better results than the radiused end mill. However, the results differ with this hypothesis. The scallop heights are larger for the radiused end mill only on small regions of the background. For the background in general, the radiused end mill still has smaller scallop height. However, at the interface of background and logo, and at the interface of background and lettering very high scallop height is observed.

A study is carried out for another sign with a complex twine border, filleted text and a star pattern background which is shown in Figure 4.11. The first stage of sign, was machined with only name, for second stage, twine border is added, and for final stage, the star pattern background is added. The sign with all the three stages are shown in Table 4.2. The results of the mean and standard deviation of scallop heights obtained for this sign is shown in Table 4.2, and the results are similar to that obtained for PBG Sign. The detailed result regarding best scallop height and best volume removal is shown in Figure 4.11. The results showed that the radiused end mill should be used to machine most of the surface. However, there are areas that remain inaccessible to the radiused cutter that can be machined by the ball nose cutter, this will always be the case as long as radius of the ball nose equals $R_1 + R_2$ of the radiused end mill. Both of the signs resulted in similar results which shows that the radiused end mills should be used for machining complex surfaces which would greatly improve the machining productivity.

4.5 Pencil Milling Technique

The radiused end milling cutter removes more volume of material but the average scallop height is higher compared to ball nose end mills. Thus, the work piece machined with a radiused end milling cutter results in lower machining time compared to ball nose end milling cutter. In order to identify the regions of high scallop height on the surface the PBG sign designed at first stage with lettering/logo. The scallop height results for ball nose and radiused end mills are shown in Figure 4.12. The highest scallop height in the radiused end milling cutter is observed when there is abrupt change in surface. In the



(a) Scallop Height Ballnose Cutter *(mm)*

(b) Scallop Height Radiused End mill *(mm)*

(c) Volume Removal Comparison d *(mm)*

(d) Maximum Volume Removal

(e) Minimum Scallop Height

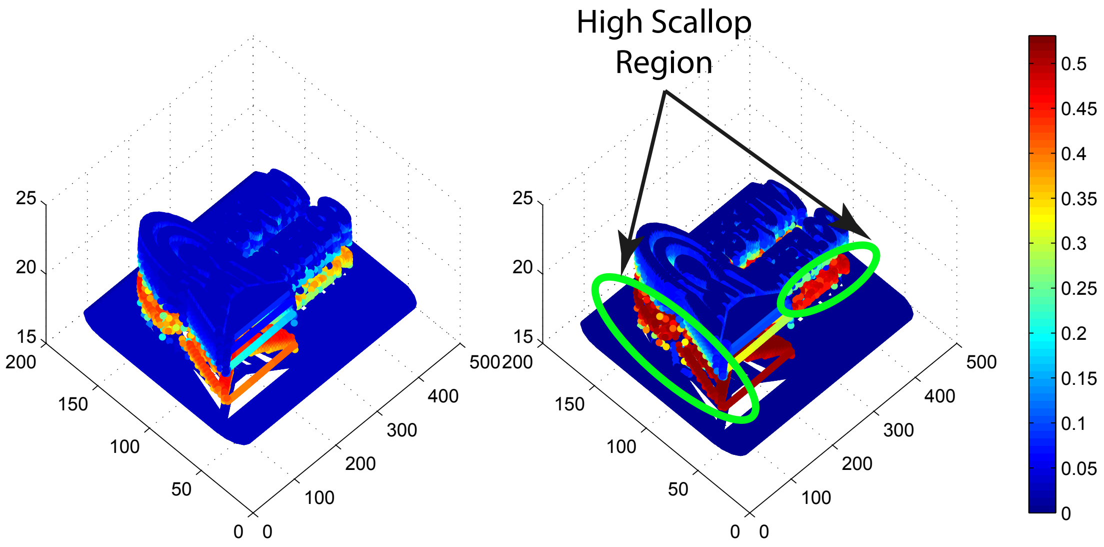
Figure 4.11: Scallop height and volume removal variation for Name Sign with text, border and background

Table 4.2: Scallop height for ball nose and radiused cutter for twine sign. Side step along and perpendicular to feed direction is $\frac{1}{32}$ inch (0.79375 mm).

	Ball Nose End Mill $R = \frac{1}{16}in$				Radiused End Mill $R_1 = \frac{1}{32}in$, $R_2 = \frac{1}{32}in$			
Type of Surface	Scallop Heights [mm]							
	Max	Min	μ	σ	Max	Min	μ	σ
Text	0.5730	0.0497	0.0553	0.0429	0.6625	0	0.0108	0.0652
Border and Text	0.5960	0.0497	0.0647	0.0676	0.6880	0	0.0436	0.1091
Background, Border and Text	0.6086	0.0232	0.0764	0.08144	0.6908	0	0.0648	0.1416
Text			Border and Text			Background, Border and Text		

wooden sign, the region of high scallop height is observed around the outline of embossed lettering and boundary of a logo. Moreover, the high scallops can be observed at the interface of logo and background, and at the interface of lettering and background. The high scallop regions while machining a sign with radiused end mills are labeled in Figure 4.12.

By comparison of Figure 4.12a and 4.12b, it can be seen that for majority of regions the radiused end milling cutter produces better results. Since the foot print along which the cutter will move remains the same, either ball nose or radiused end milling cutter can



(a) Scallop Height for Ball nose end mills

(b) Scallop Height for Radiused end mills

Figure 4.12: Scallop Height after applying variable side-step

be used. If all the regions are machined with ball nose end mills then the machining time is higher and the scallop heights in most of the regions is higher which is undesirable. If all the regions are machined with a radiused end milling cutter then the machining time would be lower but the boundary of lettering and logo have very high scallops. In order to reduce these scallop in a sign, if the side-step is reduced, the tool path length and machining time increases.

To address this issue a separate tool path with a foot print can be developed which will move the cutter across the boundary of the lettering and logo, and clean-up all the scallops around the boundary. If the interface is machined by a separate tool path, the side-step for machining can be increased to reduce machining time. The high scallops observed around the boundary of lettering and logo is reduced by developing a “Pencil Milling Technique” which will move around the boundary of lettering and logo. Thus, a part can be machined with a radiused end milling cutter within less machining time and then a separate *pencil tool path* can be generated with a ball nose cutter to remove the scallops from the region which are inaccessible by radiused end milling cutter.

In "Pencil Milling Technique", the tool path planning system receives foot print directly from the CAD model, and the tool path is generated based on that foot print which reduces loss of information regarding topology and dimensional accuracy of the surface. This "Pencil Milling Technique" which would also act as a clean-up tool path to machine the outline of letter and a logo with smaller radius tools to remove the remaining material that are inaccessible with larger tools used for previous roughing and semi-finishing tool paths. The advantage of this techniques is that it will obtain topological information during the design phase, which would eliminate the gouging and other machining errors. Moreover, since the the cutter will follow the CAD surface, any unwanted motion of the cutter is eliminated. This will reduces the tool path length and number of tool passes required to machine a sign.

This separate tool path will use the CAD model to generate tool path foot print along which the cutter will move. This technique merges the tool path planning with the CAD design i.e. from the CAD model itself an information is passed regarding the x and y coordinate of this custom foot print. The tool path planning system will generate gouge-free cutter location at each of this x and y coordinated specified using "Dropping Method". By obtaining the foot print information from CAD model and applying "Dropping Method" for positioning the cutter along this foot print, the problems observed while machining lettering using "Ball-Drop" Method can be reduced. Moreover, the problem of surface burrs discussed in Chapter 2 could be easily solved by machining the boundary of logo and lettering with a separate foot print. This would also result in better surface finish

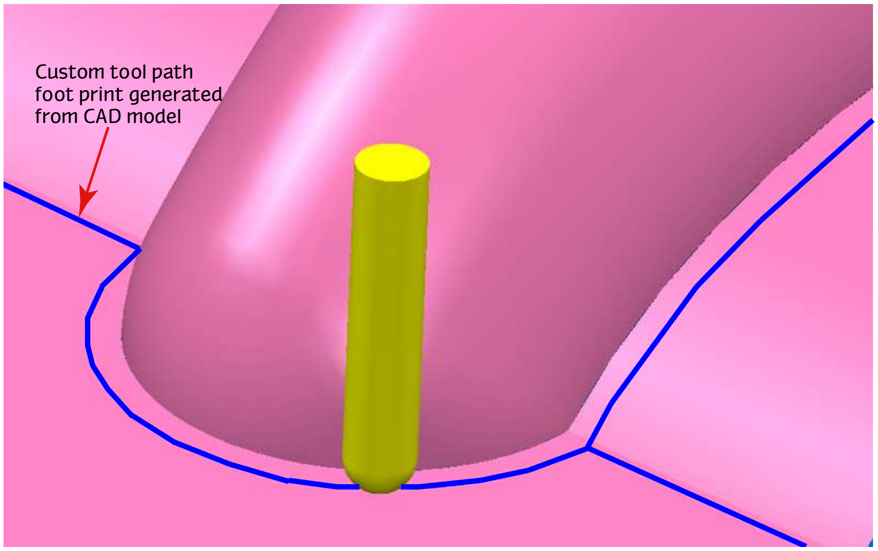


Figure 4.13: Profile tool path for machining letters

compared to the sign machine with only ball nose or radiused end mills.

An algorithm for “Pencil Milling Technique” identifies the lettering and logo from the CAD model of a sign, and generates a foot print based on their shape. While generating CAD model, a sketch of the lettering and a logo is offsetted based on the radius of the cutter R and type of cutter with which it should be machined. Thus, an offsetted boundary of the logo and a sign is obtained along which the cutter is moved. This foot print moving along the boundary of the logo and a sign is discretized in to points. The discretization of foot print into points is calculated based on the radius of the cutter R and side-step with which tool path must be generated. Each of this cutter points with its x and y coordinate with respect to origin is found and and the location of each of this points is stored in an file. This file is passed to tool path planning system to generate tool path which will move the cutter along the foot print. The gouge-free cutter position at each x, y position is found using “Dropping Method” discussed in Section 3.4. Thus, a sequence of cutter motion is obtained with just the depth of cutter(z coordinate) unknown. Using “Dropping Method”, the gouge-free depth at each of this cutter position is calculated.

An additional information regarding lifting the cutter to avoid global interference with the other features of a sign is passed in the file, to indicate that cutter must be lifted. The information regarding lifting the cutter is passed through a value *jump* and this value is stored in a file. The file passing from CAD system to tool path planning system contains information regarding values of x, y and *jump*. The tool path planning system parses this information, and drops the cutter at each x, y position mentioned in that file. The *jump* indicates that cutter must be lifted to a safe distance before moving to next position. While parsing the file, whenever CAM system finds *jump*, the x and y coordinate of the cutter is kept same, and the z coordinate of the cutter is lifted to a safe distance i.e. a distance where the cutter will not hit any feature of the work piece. The safe distance is decided based on the maximum height of the work piece. The CAM system will add a line of code in a tool path which performs this action. Thus a custom foot print can be developed by using x and y coordinate from the CAD system and calculating depth z from CAM system. By integrating CAD system with a CAM system, the custom tool path can be generated which will machine the portion of signs where the scallop height is higher.

A fully sculptured wooden sign having rectangular shape, circular border, scallop cut background, a logo and text is machined with a radiused end milling cutter which is shown in Figure 2.4. This sign was machined using $1/2^{nd}$ inch Roughing Radiused end mill, $1/8^{th}$ inch finishing Radiused end mill with a parallel spiral tool path foot print and step over distance of $1/40^{th}$ of an inch. The result of the sign is shown in Figure 4.14. The average scallop height in the whole sign machined using radiused end milling cutter shown in Figure 4.14 is found out to be 0.0711 mm (2.8008 thou). The sum of scallop heights at each consecutive cutter position (total scallop left-over) on the work piece is calculated to be 11644 mm. In the Figure 4.14, the very high scallop heights were observed

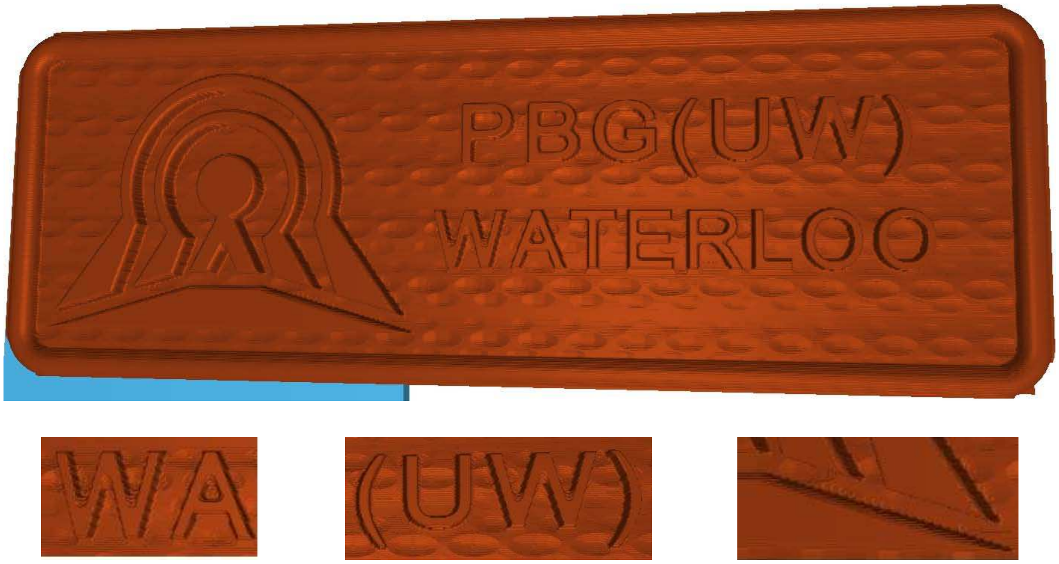


Figure 4.14: Tool path Generated before applying pencil tool path

around working around the outline of logo and at the boundary of lettering. The high scallops can also be seen in the Figure 4.12. This was reduced by applying a pencil milling technique after machining a sign described in Figure 4.14. The final sign machined with a radiused end millling cutter, and using pencil milling technique to remove scallops around the boundary is shown in Figure 4.15. By machining a sign with a bigger radiused end milling cutter and using pencil milling technique to clean-up high scallop left-over regions, surface finish and machining time can be improved. The average scallop height in the whole sign machined using radiused end milling cutter shown in Figure 4.14 is found out to be 0.03989 mm (1.5708 thou). The sum of scallop heights at each consecutive cutter position (total scallop left-over) on the work piece is calculated to be 7666.42 mm.

The total time required to machine a wooden sign was found out to be 95 minutes. There was significant reduction in machining time from 150 minutes to 95 minutes by

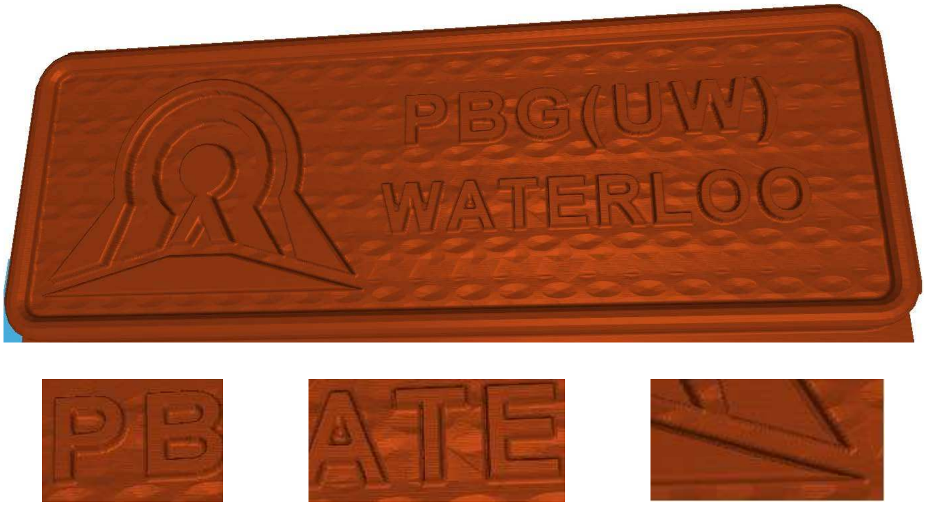


Figure 4.15: Tool path Generated after applying pencil tool path

using a radiused end milling cutter with a pencil tool path. The final sign machined on a 3 axis CNC milling machine is shown in Figure 4.16. Thus, by using a radiused end milling cutter with a bigger tool, a higher volume is removed which improves machining time, and by using “Pencil Milling Technique” to remove scallops from the regions which are inaccessible by radiused end milling cutter, the final surface finish of sculptured wooden sign is improved.

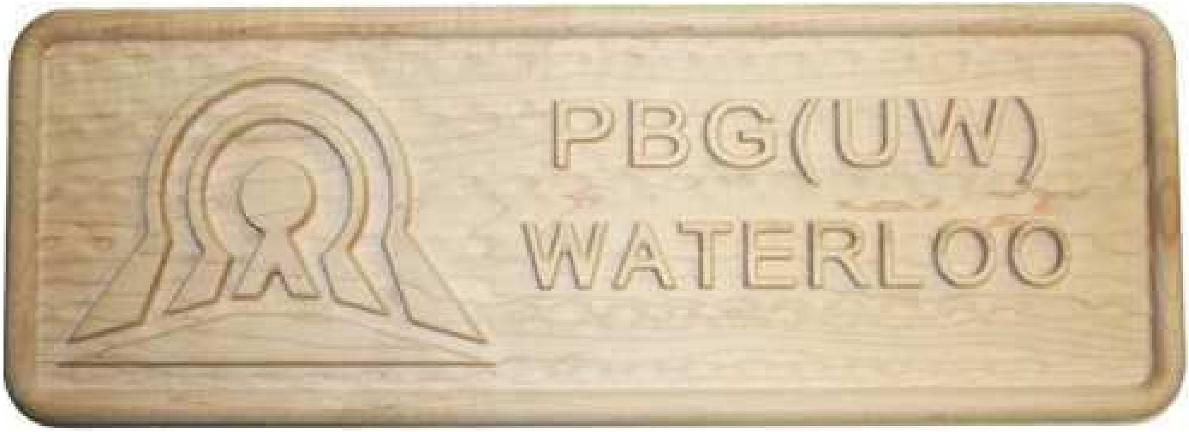


Figure 4.16: Wooden sign machined in 3 axis CNC milling machine

Chapter 5

Conclusion and Recommendation

A method for web-based automatic tool path planning for machining sculptured wooden sign is proposed in this thesis. This method generates tool path by separating tool path planning into tool path foot print and cutter positioning. By separating the foot print with cutter positioning, the tool path can be generated for any type of surface to be machined with any type of end milling cutter. The foot print for the tool path is generated at the design stage from the CAD model based on the shape of the surface being machined. The foot print generated from the CAD model, represents the shape precisely will eliminates all unnecessary cutter movement while planning the tool path. This tool path foot print is then discretized based on the tool radius and side-step to generate positions where the cutter is dropped. The cutter positioning is achieved by using "Dropping method" in which ball nose, flat-end and radiused end milling cutters are dropped on a sculptured triangulated surfaces and gouge-free cutter position is found. This method have an in-built gouge checking mechanism which will position the cutter over the triangulated surface.

The optimization strategy for lowest machining time and highest surface finish is successfully developed in this work. The approach for optimal cutter selection is used to compare results of surface finish and machining time obtained by different end milling cutter. The volume removal and the scallop height left-over methods is chosen to measure the effectiveness of result obtained from different cutters. The methodology described successfully compares a radiused and ball nose end mill based on scallop height and volume removal of each. An analysis of just the volume removal is not enough to analyze a machining process. Volume removal only determines the material removal at a certain cutter location but tools move during a machining process. It is necessary to study the effects caused by the movement of the tool.

A cutting test was conducted to study the inclination angle wrt tool axis, at which transition occur from radiused to ball nose end mills. The transition from radiused end miling cutter to ball nose end milling cutter was found at 36° for standard tooling. The performance of the radiused end mill is highly dependent on the surface geometry and it performs best when it is machining along the surface curvature. The radiused end mill provides better scallop results for the surface with lower inclination angle along feed direction compared to ball nose cutter. Similar effect can be seen for the volume of material removed for ball nose and radiused end mill, but the transition occurs at higher inclination angle compared to the transition occurred at scallop height. The radiused end milling cutter produce better results for the volume of material removed.

In case of wooden sign, the scallop heights for radiused end milling was lower compared to ball nose end mills, but in only few regions high scallop heights were observed. The highest scallops, in case of radiused end mill, are found around the boundary of the lettering and logo. To reduce high scallop heights from the region, a separate tool path is generated which will move along the profile of the lettering and logo. The separate tool path developed using Pencil Milling Technique easily removes scallops left-over with a smaller cutter from all the areas which are inaccessible by bigger radiused end milling cutter. The separate pencil tool path generated on a sign machined with radiused end mills, the average scallop height is reduced from 0.0711 mm (2.8008 thou) to 0.03989 mm (1.5708 thou). The total machining time was reduced from 150 minutes to 95 minutes for machining a 17” x 8” x 1” fully sculptured wooden sign.

Bibliography

[1] Warkentin A., Ismail F., and Bedi S. Intersection approach to the multi-point machining of sculptured surfaces. *Computer Aided Geometric Design*, 15:567–584, 1998. 31

[2] Jacob C. C., Kai G. G. K. and Mei T. Interface between cad and rapid prototyping systems. part 1: A study of existing interfaces. *International journal of Advanced Manufacturing Technology*, 13:566–570, 1997. xi, 32, 35

[3] Ju F. and Barrow G. A technologically oriented approach for the economic tool selection and tool balancing of milled components. *International Journal of Advanced Manufacturing Technology*, 14:307–320, 1998. 3

[4] Glaeser G., Wallner J., and Pottmann H. Collision-free 3-axis milling and selection of cutting tools. *Computer-Aided Design*, 31:225–232, 1999. 31

[5] Israeli Gilad. Software simulation of numerically controlled machining. Master's thesis, University of Waterloo, 2006. 81

[6] Vickers G.W. and Quan K. W. Ball-mills versus end mills for curved surface machining, journal of engineering for industry. *International Journal of Advanced Manufacturing technology*, 111:22–26, 1989. 73

[7] Zeid Ibrahim. *Mastering CAD/CAM*. McGraw Hill Higher Education, 2005. 3

[8] Redonnet J. M., Rubio W., Monies F., and Dessein G. Optimising tool positioning for end-mill machining of free-form surfaces on 5-axis machines for both semi-finishing and finishing. *The International Journal of Advanced Manufacturing Technology*, 16:383–391, May 2000. 17

[9] Patel K., Bolaos G. S., Bassi R., and Bedi S. Optimal tool selection for 3-axis milling of sculptured surfaces. *International journal of Advanced Manufacturing Technology*, 2009 (Submitted). xii, 80

[10] Tang K., Cheng C., and Dayan Y. Offsetting surface boundaries and 3-axis gouge-free surface machining. *Computer Aided Design*, 27:915–927, 1995. 31

[11] Cao Lianfang. Multi-contact tool positioning using optimization method. Master's thesis, University of Waterloo, 2000. 9

[12] Manos N., Bedi S., Miller D., and Mann S. Single controlled axis lathe mill. *International journal of Advanced Manufacturing Technology*, 32:55–65, 2007. 18, 20, 30, 31

[13] Rao N., Ismail F., and Bedi S. Tool path planning for five-axis machining using principle axis method. *International Journal of Machine Tools and Manufacture*, 37:1025–1040, 1997. 31

[14] Manos Nick. Single controlled axis lathe mill. Master's thesis, University of Waterloo, 2002. 14, 18, 19, 28, 73

[15] Gray Paul. *Region-Based Tool positioning for 3 1/2 1/2 Axis and 5-Axis Surface Machining*. PhD thesis, University of Waterloo, 2004. 20, 21

[16] Baptista R. and Simoes J. A. Three and five axes milling of sculptured surfaces. *Journal of Material Processing Technology*, 103:398–403, 2000. 31

[17] MIT Sloan Management Review. Massachusetts institute of technology smart customization group. `http://sloanreview.mit.edu/the-magazine/articles/2009/spring/50315/cracking-the-code-of-mass-customization/`, April 2009. 2

[18] Bedi S., Ismail F., Mahjoob M. J., and Chen Y. Toroidal versus ball nose and flat bottom end mills. *International journal of Advanced Manufacturing Technology*, 13:326–332, 1997. 42

[19] Mann S., Bedi S., Israeli G., and Zhou X. Machine models and tool motions for simulating five-axis machining. *Computer-Aided Design*, 42:231–237, March 2010. 81, 82

[20] Pande S. Internet based product design and manufacturing system for intelligent cnc machining. `http://webnc.cam.iitb.ac.in/WebNC/index.html`. 17

[21] Kandala Tarun. Implementation of user-defined features in web-based cad applications. Master's thesis, University of Waterloo, September 2009. 7, 10, 11, 61

[22] Huang Y. and Oliver J. H. Non-constant parameter nc tool path generation on sculptured surfaces. *International Journal of Advanced Manufacturing technology*, 9:281–290, 1994. 31

[23] Mizugaki Y., Minghui H., Sakamoto M., and Makino H. Optimal tool selection based on genetic algorithm in a geometric cutting simulation. *Annals of the CIRP*, 43:433–436, 1994. 73

[24] Choi Y. K., Banerjee A., and Lee J. W. Tool path generation of free form surfaces using bezier curves/surfaces. *Computer and Industrial Engineering*, 52:486–501, 2007. 8