2015

Computer aided process planning for multi-axis CNC machining using feature free polygonal CAD models

Ashish Mukund Joshi
Iowa State University

Computer aided process planning for multi-axis CNC machining using feature free polygonal CAD models

by

Ashish Mukund Joshi

A dissertation submitted to the graduate faculty

in the partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Industrial Engineering

Program of Study Committee:
Matthew Frank, Major Professor
Frank Peters
Iris Rivero
John Jackman
James Oliver

Iowa State University

Ames, Iowa

2015

TABLE OF CONTENTS

ACKNOWLEDGEMENT

I would like to take this opportunity to express my gratitude and thank those who helped me in every way they could towards completing this dissertation. Without their support, this would not have been possible. First of all, I would like to thank my advisor Dr. Matthew Frank. His inspiring advice has guided me throughout my research and study during past six years. He has always been kind and supportive which made my research and project-work a pleasant experience. Most importantly his work and student management nature is something I will look up to for rest of my career. I couldn't have asked for a better major professor.

I would also like to thank Dr. Frank Peters for recruiting me as a T.A. for spring 2009 semester that made it possible for me to be a student at Iowa State University. Taking his courses has always been a fun learning experience.

Next, I would like to express my sincere gratitude to my other committee members Dr Iris Rivero, Dr John Jackman, Dr James Oliver for doing me a favor by being on my committee.

I am extremely thankful to my lab mates Lei Shuangyan, Prashant Barnawal, Chen Niechen, Hou Guangyu, Zhu Siqi for making these 6 years of journey fun, enjoyable, memorable for life and full of support. It would have been difficult to remain sane without you all.

Last but not least, I owe the success of this dissertation to my family for their love and support. I would not be here today without their continued love and support.

ABSTRACT

This dissertation provides new methods for the general area of Computer Aided Process Planning, often referred to as CAPP. It specifically focuses on 3 challenging problems in the area of multi-axis CNC machining process using feature free polygonal CAD models.

The first research problem involves a new method for the rapid machining of Multi-Surface Parts. These types of parts typically have different requirements for each surface, for example, surface finish, accuracy, or functionality. The CAPP algorithms developed for this problem ensure the complete rapid machining of multi surface parts by providing better setup orientations to machine each surface.

The second research problem is related to a new method for discrete multi-axis CNC machining of part models using feature free polygonal CAD models. This problem specifically considers a generic 3-axis CNC machining process for which CAPP algorithms are developed. These algorithms allow the rapid machining of a wide variety of parts with higher geometric accuracy by enabling access to visible surfaces through the choice of appropriate machine tool configurations (i.e. number of axes).

The third research problem addresses challenges with geometric singularities that can occur when 2D slice models are used in process planning. The conversion from CAD to slice model results in the loss of model surface information, the consequence of which could be suboptimal or incorrect process planning. The algorithms developed here facilitate transfer of complete surface geometry information from CAD to slice models.

The work of this dissertation will aid in developing the next generation of CAPP tools and result in lower cost and more accurately machined components.

CHAPTER 1. INTRODUCTION

1.1 Background

1.1.1 Computer Aided Process Planning (CAPP)

The research in this dissertation focuses on Computer Aided Process Planning (CAPP) for advanced manufacturing applications. A process plan for a manufactured component can include all the steps required to deliver the completed part, from preparing the raw material, fixturing, tool and/or mold selection, setup planning, and the specific machine instruction for the actual process that creates it. Prior to manufacturing a physical product using any process, it is necessary to consider various product attributes such as geometric complexity, material, surface finish, geometric accuracy, application, etc. These product attributes influence the choice of the manufacturing process and its relevant parameters that need to be set for each product. In order to efficiently produce a high quality product, optimal process parameters must be determined through an extensive analysis of the product's critical attributes (e.g.; overall size, material, geometry, tolerances, surface finish, etc.). When this analysis is aided by the use of software, it is commonly called *Computer Aided Process Planning* (CAPP). Without CAPP, process planning must be done manually by a skilled operator, technician or manufacturing engineer skilled in the particular manufacturing process.

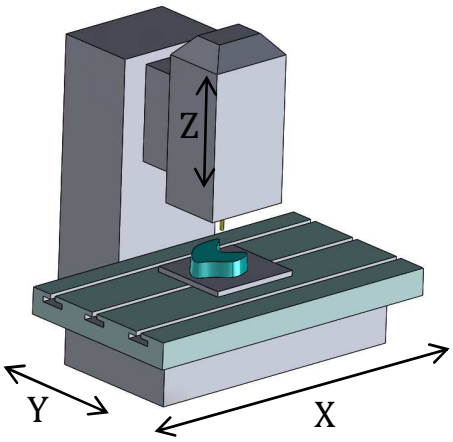


Figure 1: Schematic of milling machine

Figure 2: CNC milling machine

This consumes a great deal of time and can often lead to generally sub-optimal plans. The use of CAPP provides more optimized process planning that provide more extensive details faster and result in higher quality, lower cost products. One can argue that the use of advanced tools like CAPP can have a huge impact on the manufacturing sector, driving down costs to avoid outsourcing, improving quality to reduce warranty issues, and allowing faster product development and reducing time to market. It is in this broader area of CAPP that the current dissertation focuses; to develop new methods for the automated process planning of CNC machining using milling.

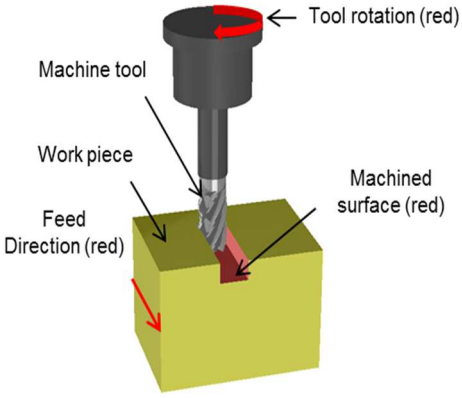


Figure 3: CNC milling process

1.1.2 CNC machining (milling)

Computerized Numerical Control or CNC was first developed shortly after WWII by John T Parson as a way to manufacture integrally stiffened skins for aircraft. Since then, CNC milling machines found applications across an array of industries and processes, where complex geometry could now be created in automated milling machines. CNC-milling is a process of incrementally cutting material from a work-piece until a pre-determined geometry is achieved, under the control of a numerically driven set of instructions executed by a computer. This involves using a simultaneously advancing and rotating tool that performs the cutting operation.

1.1.3 Multi-axis machining

The complexity of the Computer Aided Design (CAD) model of a component is directly related to the complexity of the milling machine that needs to be chosen for the machining process. Typically the manufacturing complexity of a model is determined by the material properties, surface finish, visibility or the accessibility to the part surfaces or features from multiple orientations. Hence depending on the part complexity, mills of increasing numbers of controllable axes are chosen, typically from 3 to 5 axes. A 3-axis CNC machine (Figure 4) is a typical configuration, which utilizes 3 standard coordinate axes X, Y and Z for the machining process where the cutting tool can move up or down along the Z-Axis while the machine table can move along X & Y-Axes. During the complete machining process for a part, the work-piece may be re-clamped in different setup orientations in a specific sequence. For each setup, subsets of the total

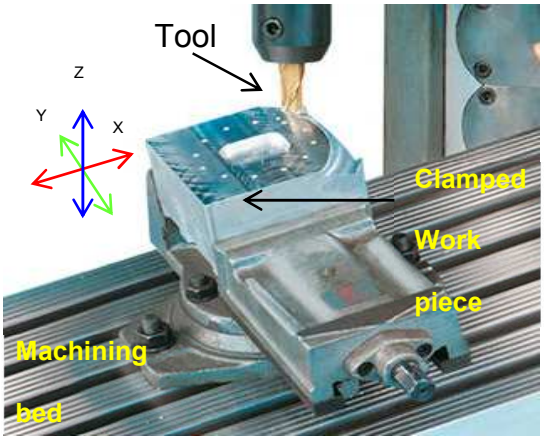


Figure 4: 3-Axis configuration

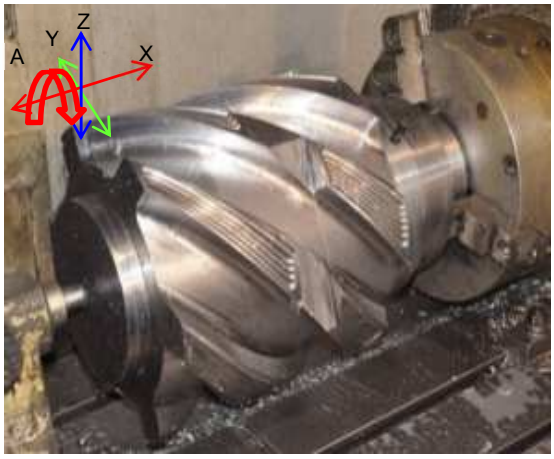


Figure 5: 4-Axis configuration

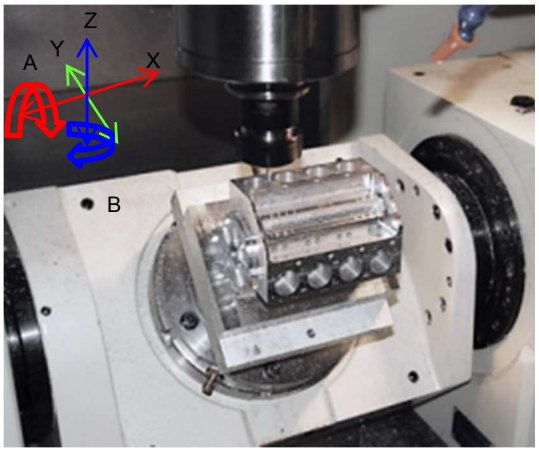


Figure 6: 5-Axis configuration

part features are generated. Some advantages of a 3-axis configuration include its cost effectiveness, relative ease of NC programming, and reduced issues with collision conditions. A 4-axis CNC machine (Figure 5) is a configuration where an additional rotary “A-axis” about the X-axis is used for the machining process in addition to standard X, Y and Z axis. In this configuration the work-piece is clamped about the rotary axis, thus eliminating the need for re-clamping of the work-piece in every setup. This can greatly reduce the time required for each setup and also improves precision by keeping the part located from setup to setup without removal and re-clamping. Lastly, the addition of a 4th axis allows nearly infinite setup orientations about the axis, making more part surfaces accessible to machining. A 5-axis CNC machine (Figure 6) is a configuration where a 5th rotary “B-Axis” about the Y-axis is used (in addition to an A-axis 4th). Five-Axis machines are commonly considered more expensive and complicated machines to use; however, they offer much expanded capability for complex geometries. Similar to a 4-Axis configuration, the part can be oriented about almost any position about a spherical space, thus reducing potential re-clamping complexities. However, with every increase in an axis for a CNC machine the NC programming becomes more complex and likelihood of failures increases. The addition of the 5th axis adds much improved flexibility, but also brings a much higher likelihood of collision conditions also, with either the spindle, tool fixture or other portions of the machine tool. Failures and collisions in 5 axis machining can be catastrophic to not only the part, but the CNC machine itself; therefore, significant care must be taken in process planning and verification.

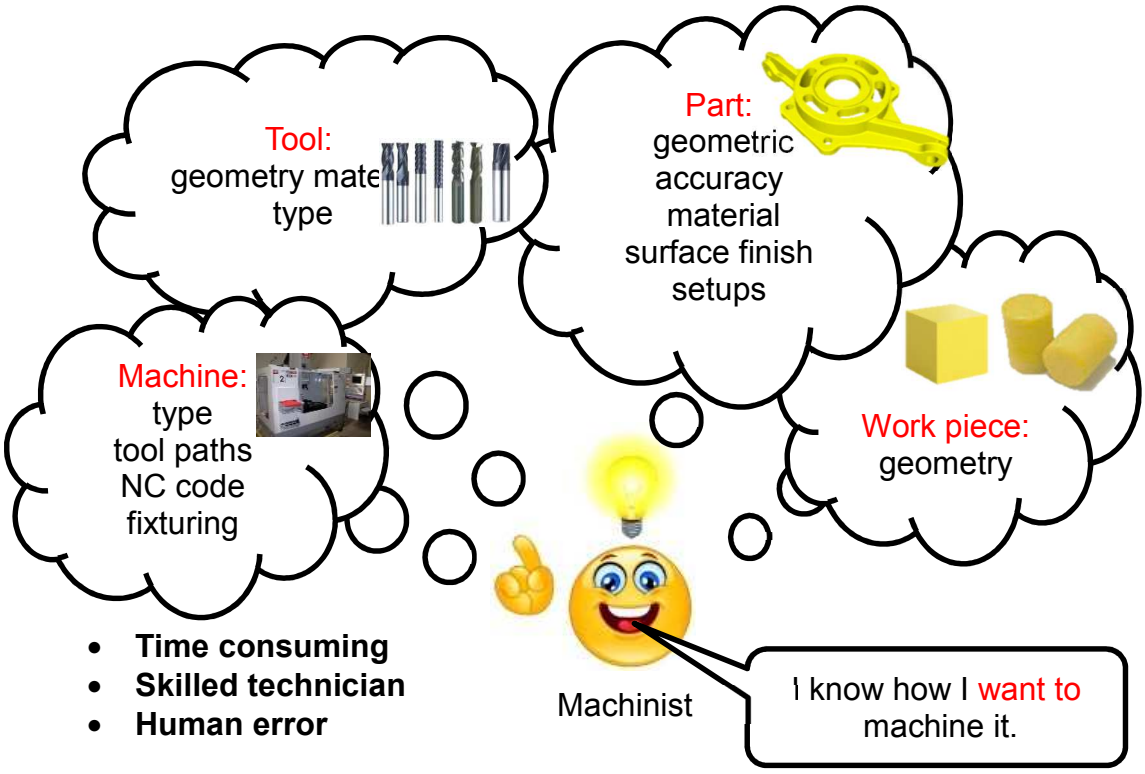


Figure 7: Manual Process Planning

1.2 Motivation

1.2.1 Process planning for CNC machining

In order to machine a component in a geometrically accurate and efficient manner, it is necessary to analyze the part and perform process planning for each step, setup and portion of machining code. Process planning must consider numerous attributes such as geometry, material composition (single/multiple), dimensions, tolerances, work piece geometry, clamping mechanism, available tool geometry, tool material, tool type, etc. It is also required to choose suitable machines and related parameters such as machining feed and speed. Finally, the machinist must generate an NC program for the machine such that the cutting tool follows a specific path and performs a series of cutting operations. When the machinist performs all the process planning tasks without assistance from automated systems such as computers, this planning is termed as

manual process planning (Figure 7). Manual process planning requires considerable skill, and often consumes a large amount of time that can slow the overall production schedule and increase costs. In the past, process planning was performed manually by analyzing 2D part designs made on blue prints by experienced designers and then NC code was written. However, advances in Computer Aided Design (CAD) systems have provided an opportunity for part designers to create intricate geometric designs effectively and quickly. By the late 20th century, this led to the idea of Computer Aided Process Planning (CAPP).

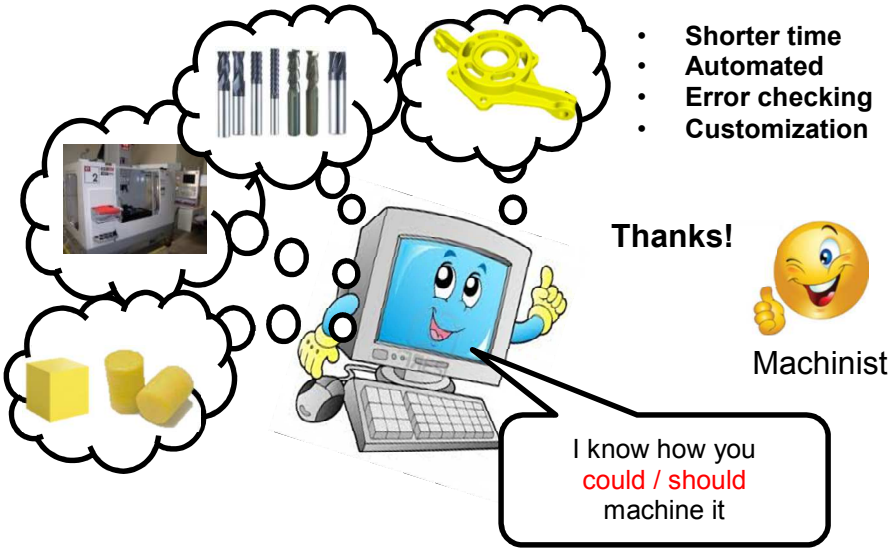


Figure 8: Computer Aided Process Planning

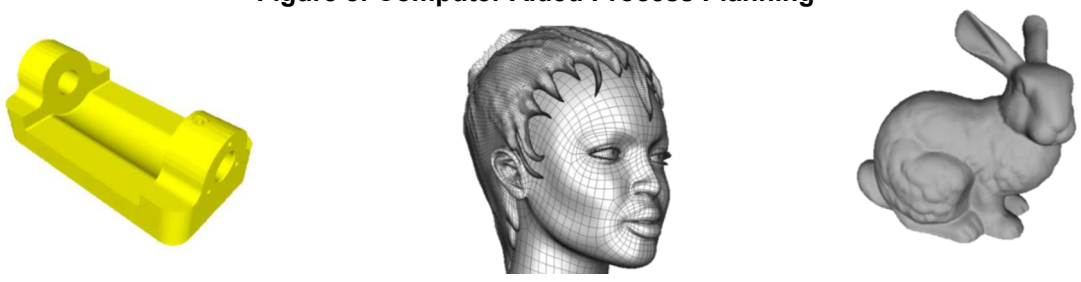
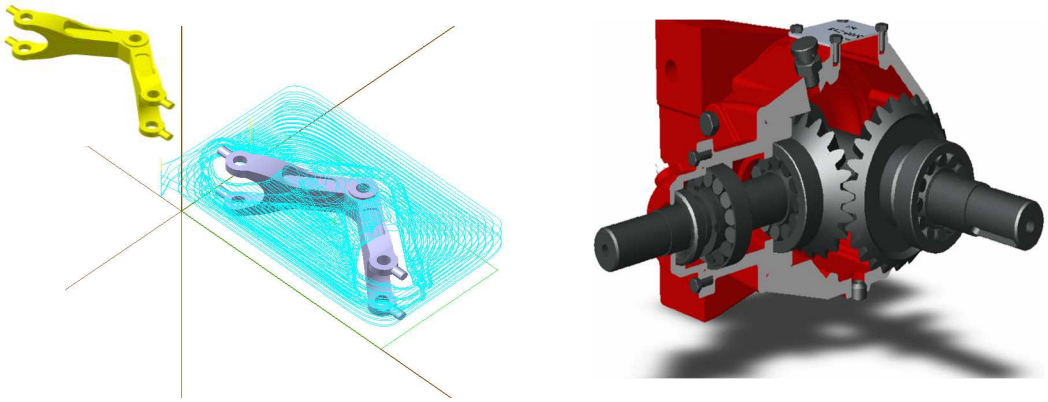


Figure 9: a) Designed prismatic b) Designed free-form c) Natural free-form

Computer Aided Process Planning has been at the forefront for providing automated solutions to process planning challenges that exist in Computer Aided Manufacturing [1-5]. These systems, through the input of CAD models, generate a process plan automatically and effectively within a relatively short time (Figure 8). This has allowed manufacturing industries to maintain advanced production schedules, improve part qualities and eliminate losses and costs resulting from inefficient manual process planning. This improvement can be seen in the form of the new customizability we find with many of today's products, in addition to the great improvements in efficient use of materials, aerodynamics, safety, etc. To this end, developments in CAD/CAM and CAPP will continue to play a significant role in the support of advanced manufacturing in our modern society. In order for CAPP systems to operate, one must have a computer model (CAD model) representation of the part design. The type of CAD models available can have a significant impact on the process. The following section will provide a summary of CAD formats in use today.



(a) (b)

Figure 10: a) CAD models for CAM b) Designed CAD

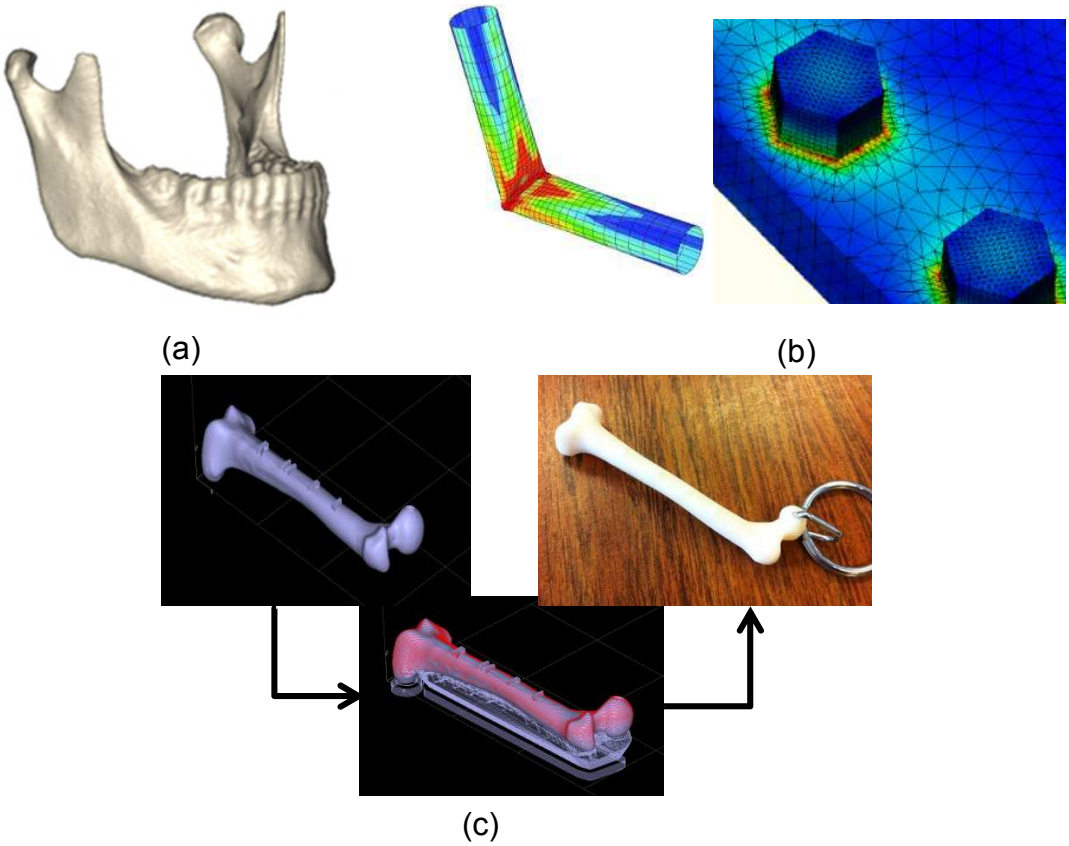


Figure 11: a) Reversed Engineering b) FEA c) CAPP

1.2.2 CAD formats

CAD models are geometric software representations of objects which may contain one or a variety of shapes, including a designed prismatic shape (Figure 9), a designed free-form shape, or naturally occurring free-form shape (Figure 9). Since the invention of geometric modeling, several mathematical formats representing model geometries have been developed by the CAD community for various engineering purposes. These formats have been used for various purposes (Figure 10,11) such as engineering design, reverse engineering (RE), graphics visualization, engineering analysis, computer aided manufacturing (CAM), and computer aided process planning (CAPP).

Geometric CAD formats describe various attributes such as surface geometry description, features, and volumetric properties that are used in CAPP. Given a CAD model, these attributes are analyzed algorithmically to generate a process plan for the efficient manufacturing of the model. Although many classifications can be used, in this work, we will categorize models as being either *feature-based* or *feature-free*. Feature-based model formats are those most commonly used in the manufacturing field.

Figure 14: Parametric free form shapes

"Features" are generic classes of shapes on a product with which the designer usually intended some function, and they have certain attributes that can be useful for reasoning in process planning (Figure 12). Some examples of engineering features include holes, pockets, slots, chamfer, fillets, etc. In manufacturing using feature based models, features are considered alone or sometimes categorized into groups for CAPP. Ultimately, the aim is to generate a process plan that would manufacture the part accurately with every feature created as designed in a cost efficient manner.

Feature-free CAD models can be placed in three common classes; 1) Voxel models, 2) Parametric models, or 3) Polygonal models. Voxel models (Figure 13) are specifically designed in a way so as to primarily represent volumetric properties in 3D space in addition to the surface. These volumetric properties could be mechanical properties

such as toughness, density, porosity, elasticity, etc. The level of granularity for these properties depends on the voxel size. A Voxel, whose term is derived from “volume” and “pixel”, is a cube shaped structure representing a value of a regular grid in 3D space (Figure 13). Common uses of voxel models include volumetric visualization in medicine, video games, terrains maps and simulations. Parametric CAD models describe surface geometry defined by mathematical curves and surfaces commonly known as Non-Uniform Rational B-Splines (NURBS) defined by parametric equations. These surfaces define the entire surface geometry of the CAD models in such a way that every geometric region or a feature on these models is approximated by surface patches (Figure 14).

Polygonal CAD formats are one of the simplest surface representations used in the approximation of free form shapes (Figure 15). The surface geometry of polygonal models is approximated by planar polygons. Though there is no limit on the number of sides a polygon may have, triangular facets are most commonly used. Polygonal models using the STL format (Figure 15) was first developed for the additive manufacturing process called Stereolithography. Using STL format, models can be

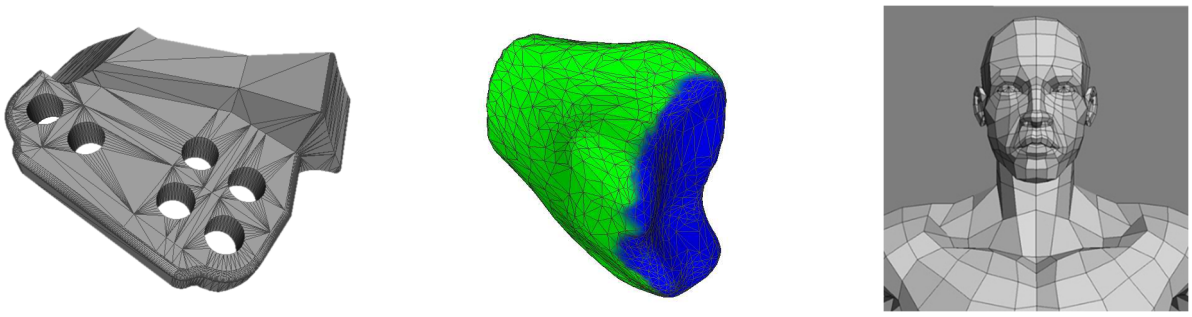


Figure 15: Polygonal free form shapes

manufactured with reasonably accurate surface geometry. However, one of the limitations of the STL format is that it is incapable of storing surface attributes like

roughness, color, hardness etc. In order to eliminate these shortcomings several other polygonal *multi-attribute* formats such as OBJ, VRML, PLY and AMF were developed. However, for these *multi-attribute* formats, CAPP methods need to be developed that would enable automated manufacturing of these part models, considering their varying attributes. Currently, most CAPP systems plan for automated manufacturing of these part models with focus on creating accurate geometry only. This thesis presents solutions to CAPP challenges that occur when multi-attributes on feature-free models have to be considered in process planning, in addition to the focus on creating accurate geometry.

Figure 16: Mechanical components

1.2.3 Application of CAPP for CNC machining

Figure 17: Bio-medical components

Parts produced by CNC machining have applications in different fields ranging from mechanical to bio-medical. Some examples of parts include machined gears, pistons, aero-foils, etc. that are used in the automotive, aerospace, and other mechanical industries (Figure 16).

Additionally, CNC machined parts are used for producing bio-medical equipment such as dental and orthopedic implants (Figure 17).

Due to the demand for higher part quantities, better quality and lower cost, it is necessary that these components are produced using optimized process plans. This is motivation for using CAPP, where one can more readily program complex plans for even the most complex geometries.

1.3 Objective

Considering the impact of Computer Aided Process Planning on manufacturing, the overarching goal of this research is to develop new CAPP methods that would allow multi-axis CNC machining of parts using polygonal CAD models. In order to address this overarching objective, this dissertation has been divided into 3 sub-objectives that will contribute to the state of the art in CAPP. The sub-objectives are as follows:

1) The first sub-objective is to develop CAPP algorithms for the rapid of machining of multi-surface Parts. The CAPP algorithms developed in this problem are demonstrated for the rapid machining of implants with preserved functionality and desired surface characteristics.

2) The second sub-objective is to develop CAPP algorithms for discrete 3-axis CNC machining of part models using feature free polygonal CAD models. This work will create a generic and cost effective 3-axis CNC machining process for which CAPP algorithms are developed. These algorithms would allow rapid machining of a wide variety of parts with higher geometric accuracy by enabling access to visible surfaces through the choice of appropriate machine tool configurations.

3) The third sub-objective is to develop solutions to the problems of unique geometric situations existing on non-feature based CAD models. The 2D slice models derived from feature free polygonal models can be used for CAPP in multi-axis machining. The conversion from CAD to slice model results in the loss of model surface information, the consequence of which can lead to suboptimal or incorrect CAPP solutions. The algorithms developed in this area will facilitate transfer of complete surface geometry information from CAD to slice models for analysis. This allows for correct and optimal CAPP solutions for multi-axis CNC machining.

1.4 References

[1] Weill, R., G. Spur, and W. Eversheim. "Survey of computer-aided process planning systems." *CIRP Annals-Manufacturing Technology* 31.2 (1982): 539-551.

[2] Steudel, Harold J. "Computer-aided process planning: past, present and future." *THE INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH*22.2 (1984): 253-266.

[3] Ham, Inyong, and Stephen C-Y. Lu. "Computer-aided process planning: the present and the future." *CIRP Annals-Manufacturing Technology* 37.2 (1988): 591-601.

[4] Alting, Leo, and Hongchao Zhang. "Computer aided process planning: the state-of-the-art survey." *The International Journal of Production Research*27.4 (1989): 553-585.

[5] Wang, Hsu-Pin, and Jian-Kaing Li. "Computer-aided process planning."*Advances in Industrial Engineering* 13 (1993).

[6] Xu, Xun, Lihui Wang, and Stephen T. Newman. "Computer-aided process planning–a critical review of recent developments and future trends."*International Journal of Computer Integrated Manufacturing* 24.1 (2011): 1-31.

CHAPTER 2.LITERATURE REVIEW

2.1 Background

Computer Aided Process Planning (CAPP) has been recognized as a key contributor to the area of Computer Aided Manufacturing (CAM), with significant impact on the part quality, quantities, reduction in manufacturing time and cost, while reducing manual intervention. Since the 1970's, tremendous effort has been put forth in developing CAPP systems for various manufacturing processes [1][2][3]. The literature review in this section provides comprehensive details of research in different areas of CAPP specifically for CNC milling, which has motivated and inspired the author for solving related challenging problems and developing this dissertation.

CAPP for CNC machining has played a pivotal role in developing efficient CNC machining processes. Automated process planning solutions for CNC machining has resulted in the production of higher quality parts while reducing the time and cost of manufacturing. Some of the research areas towards CAPP for CNC machining have been topics such as automated fixture planning, setup planning considering part visibility, and accessibility using different tool configuration, tool path generation, tool selection, machining parameters selection [1-10], application of advanced computer systems such as artificial intelligence [11], using advanced hardware such as the Graphics Processing Unit (GPU) [12-13], etc. The literature review in this section provides a comprehensive overview of different areas in CAPP for CNC machining considering widely used CAD formats. At the end of literature of review, the author presents an argument for choosing to develop new methods for setup planning for multi-axis CNC machining using feature free polygonal models.

2.2 Process Planning Using Various CAD Model Formats

Performing CAPP for CNC machining processes requires extensive analysis of the CAD models that are to be machined. Since the invention of Computer Aided Design, significant contributions have been made by developing various geometric formats. These formats may have been developed for specific purposes such as Manufacturing, Engineering design, Analysis, Visualization, Reverse Engineering, Inspection etc. Previously for the sake of manufacturing industrial parts, 2D blueprints used to be drafted by design engineers. However these 2D designs were difficult to interpret and made the overall process planning slow and costly. The development of CAD systems has made it easier for the designers to create 3D models with complicated features and surfaces satisfying complicated functionalities. Hence In this subsection, review is presented on CAD models divided primarily into 2 categories; 1) Parametric, and 2) Polygonal models and provides a review of various process planning methods using above two said CAD model formats.

CAD models that are designed primarily from parametric mathematical curves and surfaces are called parametric models [14]. Parametric models are commonly designed from the class of curves and surfaces called as Non-Uniform Rational B-Splines, commonly referred to as NURBS. NURBS provide a unified mathematical basis for representing both analytic shapes like conic sections and quadrics surfaces as well as free form surface entities like car bodies [14]. Hence the parametric models can be further divided into 1) Feature based models and 2) Feature free models.

Feature on a product is considered as the geometric shape or characteristic with which certain attributes and knowledge is associated which could be useful for reasoning

about that product [15][16]. One of the commonly used feature based formats is STEP, which is popular and widely used Computer Aided Process Planning in manufacturing. In addition to the NURBS data this format can also hold other manufacturing requirements data such as Geometric Dimensioning & Tolerancing (GD&T), surface finish, or other manufacturing specific properties. Xu et al [17] presented a comprehensive review of STEP-NC developments and their futuristic applications for CAD, CAPP, CAM and CNC integration.

Fixture planning is an important part of CAPP for the CNC machining process. Optimal and efficient fixturing is always required to clamp the part with high stiffness such that it deflects minimally and is machined within required tolerance specifications. Several key research contributions have been done in the area of fixture planning using feature based CAD formats. Bi [18] et al, Kang et al [19], Hargrove et al [20] presented an extensive overview of computer aided fixture design. Traditional fixturing techniques involve use of hardware such as clamps, vises, V-block, modular plates, etc. This may introduce several issues such clamping errors, limited tool access due to clamps, increased setup complexity and time. Therefore it is always required to extensively analyze a part for fixture design in order to increase part accuracy. Dong et al [21] investigated the use of features for fixture design, concentrating on the selection of locating elements and the identification of locating surfaces for work-piece positioning. Zhou et al [22] proposed a feature based fixture design methodology in which previous fixture design cases and design rules are described in association with features and thus the design knowledge is integrated with geometric information. In this methodology, machining features of the structural parts and their associated attributes

are identified by feature recognition technique from 3 dimensional (3D) part models defined by model-based definition (MBD) technique. The feature-based part information models are then established and used to retrieve previous fixture design cases and design rules stored in the knowledge base. Fixture designers will choose the appropriate retrieved design cases as the starting point for new fixture designs or use previous designs to assess his/her current design. Perremans et al [23] reported on the development of an expert system for the design of the physical fixture for prismatic parts on machining centers. After inputting the faces on which positioning, clamping and/or supporting should be done, the expert system generates the necessary assembly of modular fixturing elements (i.e. the physical fixture). New in the approach is the description of modular elements by means of form features. This method permits the description of an arbitrary modular fixturing system with the only restriction that it has to be based on a grid of holes.

Tool path planning is another important area of process planning, wherein numerically created toolpaths are used to precisely shape the work piece based on the part geometry. Although manual tool path planning is feasible for prismatic and small parts, when parts are complicated or free form in nature it becomes necessary to design more complex paths. Additionally, as the number of axes in the CNC machine increase the concerns for tool collision increases significantly. Tool path have also been developed considering different types of CAD models where some of them fall into feature based and others into feature free types. This has lead the CAPP research community to develop advanced, optimal, safe and economical tool paths for various types of CAD models. Dragomatz et al [24] presented a classified bibliography of the literature on the

NC milling path generation. In one of the works for pocket milling, the tool path problem was formulated as a Travelling Salesman Problem (TSP). In [25] The pocket area was digitized into a finite number of tool points with the development of a new neural network method and development of post processing for path smoothing and feed rate adjustment. Held [26] presented a zig-zag algorithm for pocket machining. This algorithm was capable of computing correct zig-zag tool paths for multiply connected planar areas. Further optimizations with respect to geometrical and technological objectives were included to achieve near optimum inclination of tool path.

In addition to toolpath planning, tool selection is a very important process planning task for CNC machining. The choice of tool configuration can have a defining effect on the accuracy of the part geometry, surface finish, tool wear, tool path efficiency and machining time. Depending on the tool geometric parameters such as profile, diameter and length, there can always be different type of tools required for different milling processes, including flat-end milling for roughing or ball-end milling for achieving high surface finish. Veeramani et al [27] described a new method for choosing optimal tool sets for 2½-D pocket milling in two phases. In the first phase, a new concept of Voronoi Mountain was used to calculate the material volume that can be removed by a specific cutting tool size, the material volume that will subsequently remain to be machined, and the cutter-paths (and corresponding processing times) for each cutting tool. Then a dynamic programming approach for optimal selection of cutting tool sizes was used. Computational experiments showed substantial savings in processing time by using multiple cutting tool sizes to machine 2½-D pockets. Hinduja et al [28] proposed a tool selection method for 2½-D pocket machining where tool diameter was chosen following

a consideration of the variation of width of cut divided by cutter diameter. It was shown that, while a smaller cutter diameter gave a favorable variation, it lead to longer tool paths. Hence the final cutter diameter was a compromise between the increased costs of a larger diameter with a shorter tool path and the lower costs of a small diameter with a longer tool path.

Setup planning in CAPP for multi-axis machining has been one of the most important and challenging areas. It provides solutions for machining orientations based on parameters such as visibility (line of sight), reachability (tool length), accessibility (tool diameter), tolerances, and multi-axis machine configurations. Diverse algorithmic approaches using mathematics, computational geometry, computer data structures etc. have been employed in order to provide automated setup planning solutions. For feature based process planning, feature based models were used for setup planning by Liu et al [29]. In their work, prismatic machining features were grouped into categories using a knowledge based approach, geometric reasoning, and machining precedence constraints. Geometric reasoning was also applied in order to sequence features for machining from a specific setup for a tool approach direction. In another feature based work for setup planning, Zhang et al [30] proposed using a hybrid-graph approach where faces of the part were considered as graph vertices (V) while the critical tolerances represented edges (E). Using this terminology, algorithms were developed to perform operation sequencing in order to derive a minimum number of setups while preserving the design tolerances. Wu et al [31] used a feature based approach for prismatic parts where tolerance specifications in a feature based design system were used to generate setup plans with explicit datum elements. The generated setup plans

were ranked based on setup count and an estimated accuracy of resultant dimensions. Ong et al [32] developed a fuzzy set theory method for determining setup orientations. The developed system formulated setup plans based on the initial, intermediate and final states of the part while considering production rules and object representations. Hebbal et al [33] developed a method to select an optimal setup plan for machining features of a given prismatic part. This work identified groups of features that could be machined in single setups, determined a suitable datum plane for each setup, then it determined all the feasible setup plans to machine the given set of features of prismatic parts. Finally, a feasible setup plan was developed based on tolerance and economical limiting conditions. Sakurai [34] developed algorithmic and heuristic methods to synthesize and analyze setup plans and fixture configurations using toleranced solid models of the finished component. Huang [35] developed setup planning algorithms for lathe machines, where algorithms included considerations for tool approach directions and tolerances represented by a graph. Chen et al [36] proposed a new approach for setup planning of prismatic parts using “Hopfield neural net coupled using Simulated Annealing”. In this work, they used two stage planning where the first stage was used to sequence all the features of a workpiece according to geometric and technological constraints while in stage 2 setups were identified from sequenced features and the precedence and the critical tolerance relationships between features were treated as constraints. Another contribution came from Sarma et al. [37] where they developed setup planning algorithms for simply fixturable components. In this work a robust graph-theoretic model of planning was presented along with hierarchical prioritization of the

objectives in planning. Some of the objectives chosen for hierarchical use were tool changes, feature intersections and feature ordering.

Kumar et al [38] developed a feature recognizer for the machining features represented in a CAD model. Following that, a rule/object-based approach was used to group the machining features into appropriate fixture setups, and to recommend suitable clamping, locating and supporting points. The appropriate fixturing elements were then selected and assembly sequences were planned. Lee et al [39] used an approach for analyzing fixture kinematics, clamping forces and clamping friction. Since many fixturing arrangements rely on friction to hold the part, it was deemed important to study friction for fixture planning.

Lin et al [40] utilized a real-time approach for 3-D parametric surface machining on 3-axis CNC machine tools. This real-time approach, called the CNC Surface Interpolator, read surface g-codes and performed surface machining interpolation. The input to the interpolator was the surface g-code, which contained geometric information, such as the coefficients of the parametric surface, as well as cutting conditions. This resulted in better machining feed rate controls, more precise machining commands, and required less machining time compared to that produced by off-line approaches. Lee [41] presented a new approach to 5-axis NC tool path generation for sculptured surface machining. Techniques for feasible machining strip evaluation were used for non-isoparametric 5-axis tool path generation. A search algorithm was proposed to find the parameter increments of adjacent cutter locations along orthogonal path intervals for optimal non-isoparametric path generation. The proposed methodology included three steps: (1) evaluating feasible machining strip, (2) solving parameter increments ($\Delta\mathbf{u}$, $\Delta\mathbf{v}$)

along orthogonal path intervals, and (3) searching for adjacent non-isoparametric cutter locations. Chen et al [42] developed a real-time CNC tool path generation algorithm for machining IGES surfaces. IGES-based CAD data files could be directly fed to the CNC machines and the tool paths generated in real time could be passed to a motion controller during cutting via a multibus II backplane structure. The real-time NC path generation algorithm could properly deal with issues such as trimming lines, gouging detection, and adaptive tool step adjustment. Elber et al [43] developed an algorithm to adaptively extract isocurves for rendering and milling toolpaths were enhanced for models consisting of trimmed surfaces for both 3- and 5-axis milling. The resulting toolpaths were shorter and provided a direct quantitative bound on the resulting scallop height while avoiding gouges.

Yang et al [44] presented a systematic tool-path generation methodology which incorporated interference detection and optimal tool selection for machining free-form surfaces on 3-axis CNC machines using ball-end cutters. In this method, the global and local interference was first detected and prevented, and then optimal tools in terms of machining time were selected and tool paths generated. A system of algorithms were developed to determine the interference area, and the machining time for each available tool was estimated by considering tool size, scallop height, and accessible surface area. Jensen et al [45] presented an automatic cutting tool selection methodology for five-axis finish surface machining based on the techniques of curvature matched machining. The criterion for cutter selection was to minimize the machine errors and to maximize material removal rate using an optimal filleted-end mill selected from a standard cutting tool library. Tool parameters investigated included cutter radius, cutter corner radius and

cutter length. Algorithms for detection and correction of local tool gouging and global tool interference were presented. Global tool interference detection and correction was solved by studying the shortest distance between the part surface and the cutter body axis and a separate approach was used to accelerate the distance calculations. Chen et al [46] proposed an optimal approach to select multiple tools for aggressive roughing of the pocket. First, the NC paths of a specific tool were quickly generated using the pocket's medial axis transform. Then, an optimization model of selecting multiple cutters and generating their NC paths is built in order to achieve the highest efficiency of the aggressive rough machining. Han et al [47] presented new method for optimizing the plunge cutter selection and tool path generation in multi-axis plunge milling of free-form surface impeller channel to improve the efficiency in rough machining.

There has been a considerable amount of work done in setup planning using parametric feature free models. Woo [48] developed visibility maps and spherical algorithms using parametric surface models which focus on multi-axis automated machining. Kai et al [49] presented methods to minimize the number of setups in a 4-axis or a 5-axis NC machine. The problem was formulated such that given a set of spherical polygons, find a great circle or a band containing a great circle that would intersect the polygon manually. Attila [50] presented a program that bridged the gap between the CAD models and setup and fixture planning, to make the planning of workpiece fastening in case of box-shaped parts easier and faster.

Polygonal mesh models were first introduced with a focus on Rapid Prototyping applications such as additive manufacturing. However they were eventually adopted for

Computer Aided Process Planning for subtractive manufacturing processes such as CNC machining.

Wu et al [51] presented a fundamental study of automated fixture planning with a focus on geometric analysis using polygonal models. The initial conditions for modular fixture assembly were established together with geometric relationships between fixture components and the work piece to be analyzed. Brost [52] presented an algorithm that accepted a polygonal part shape as input and synthesized the set of all fixture designs that would achieve form closure for the given part. Boonsuk et al [53] developed automated sacrificial fixturing methods for a rapid CNC machining process using slice models generated from a polygonal model. As compared to the traditional fixturing hardware, the sacrificial supports emerge incrementally at the end of the machining process where these elements connected the part to the remaining stock model.

Polygonal mesh models have also been extensively used for tool path planning for CNC machining. Bertoldi et al [54] used domain decomposition to divide arbitrary layer geometries into smaller regions of simpler shapes. Yuwen et al [55] presented a new approach to iso-parametric tool path generation for triangular meshes. The strategy proposed the parameterization of the triangular faces via harmonic mapping. The cutter-contact (CC) points and the path interval were then calculated based on the machining tolerance requirements and the iso-parametric tool paths were finally generated. Park et al [56] presented an optimized procedure for tool path generation in regional milling. The proposed procedure computed tool paths by slicing a cutter location surface, which is a triangular mesh containing valid triangles. Chen et al [57] developed path planning for 3-axis ball milling of polygonal free form surfaces. In this work the vertices of

triangles were offset along normal and an offset mesh was created and sliced along an axis to obtain parallel tool paths. Lee et al [58] developed tool path using mesh surfaces for constant scallop height. Zhang [59] presented an efficient greedy strategy for generating tool paths on triangular meshes with consideration of axis kinematics. Bi et al [60] presented a Graphics Processing Unit (GPU) based approach that generated a collision free and orientation-smooth tool paths for five axis NC finish machining using a ball end cutter. Balasubramaniam et al [61] described a system that generated 5-axis roughing tool paths directly from a tessellated representation of a body. Here the tool paths were directly generated from the shape of the work piece using measures of accessibility to avoid collisions.

Different tool selection strategies considering visibility and accessibility have been developed using free form polygonal models. D'Souza [62] described an efficient method to find the lowest cost tool sequence for rough machining free form pockets on a 3-axis milling machine. The free form pocket was approximated to within a predefined tolerance of the desired surface using a series of 2½-D layers of varying thicknesses that could be efficiently removed with a flat end milling cutter. A graph-based method was used to find an optimal sequence of tools for rough machining the approximated pocket.

As compared to feature based models and feature free parametric models, some amount of research has also been performed in setup planning using feature free polygonal models. Feature free polygonal CAD models are those that represent accurately the surface geometry of the part using planar Euclidean polygons without the capability for recognizing geometric features. This is particularly useful when the parts

that are to be machined are free form in nature, for example, car bodies, sculptures, bone implants, or any naturally occurring shape or models generated by reverse engineering. Since their conception, there have been some significant works related to setup planning using polygonal models for CNC machining. Frank [63] used slice models generated along an axis from a polygonal model and designed visibility algorithms about the slicing axis. The results from the visibility algorithms could be further useful towards determining setup orientations for multi-axis machining. Li and Frank [64] presented a feature-free method for determining feasible axes of rotation for setup planning, based on the visibility of a polyhedral model. Li and Frank [65] also presented tool accessibility algorithms using slice models derived from feature free polygonal models. Li and Frank [66] focused on determining non-visibility cones, which are the complementary sets of visibility of convex polygonal facets. Their approach evaluated the boundaries of the non-visibility cones of an arbitrary convex polygon due to the visibility blocked by an obstacle polygon. The intent of this work was to develop a feature free approach to setup planning with focus multi-axis machining setup. Visibility maps from polygons [66] could provide a quantitative evaluation of a surface, a feature or an entire part model. However the next step was to use the visibility information for setup planning. Specifically algorithms were developed for indexer type machining setup. Due to the use of polygonal visibility, feature detection was not required. Haghpassand and Oliver [67] formulated an optimal design problem based on a discrete approximation of design surface geometry, the kinematic capabilities of the process machine tool and processing cost. The problem was formulated as a constrained and nonlinear optimization problem for both 3- and 4-axis machining. Dhaliwal et al [68]

described algorithms for computing global accessibility cones for each face (i.e., the set of directions from which faces are accessible) on a polyhedral object. Exact mathematical conditions and the associated algorithm were presented for determining the set of directions from which a planar face with triangular boundary is inaccessible due to another face on the object. Spyridi and Requicha [79] presented algorithms for computing Global Accessibility Cones for faces for polyhedral solids. This also included calculating “silhouettes” of solids generated by Minkowski operations. Vafaeesefat and ElMaraghy [70] presented a method for determining optimal work piece orientation for sculptured surfaces for 3-axis machining process application. Finally Lasemi et al [71] have presented a detailed state-of-the-art review of Computer Aided Process Planning methods for CNC machining of free form surfaces.

2.3 Motivation

Considering the above reviewed work, it has come to light that no known practical purpose CAPP systems exist that would allow a machinist to input polygonal and multi-attribute models and perform CAPP for multi-axes CNC machining. For CNC machining, CAPP solutions using polygonal formats should include automated setup planning, fixture planning, tool selection, and mesh processing. Setup planning is one of the most challenging steps that a machinist has to carry out in order to manufacture a part. Once the setups are determined, fixture planning and tool selection could follow. In the absence of an automated setup planning system, the machinist will analyze the CAD geometry visually and determine setup solutions manually which may or may not be correct or optimal. Hence, this dissertation focuses on designing algorithms that are implemented for automated setup planning systems for CNC machining using polygonal

and multi-attribute CAD models. The layout of the remainder of dissertation is as follows. In chapter 3 the author proposes setup planning algorithms for CNC machining of multi-colored polygonal models. Depending on the application, the different colored regions may represent different surface regions or features. In chapter 4, the author presents new setup planning algorithms using uncolored polygonal models for multi-axis machining processes, theoretically without limit on number of axes. This challenge is classified into an “embarrassingly parallel” category which has allowed the use of parallel processing hardware GPU for achieving better and quicker setup planning solutions. The potential applications to the proposed algorithms in chapter 3 and 4 could be used for the CNC machining of a variety of industrial or biomedical components of varying complexity. In chapter, 5 the author presents a unique solution to a condition caused by the slicing of polygonal models; since the slicing occurs along an axis, polygons parallel to the plane are never sliced. The author provides a solution to this condition that could significantly improve process planning. The solutions from chapter 5 are actually integrated into chapter 3 and 4 such that the process planning carried out in each section is complete and thorough.

2.4 References

[1] Xu, Xun, Lihui Wang, and Stephen T. Newman. "Computer-aided process planning–A critical review of recent developments and future trends."*International Journal of Computer Integrated Manufacturing* 24.1 (2011): 1-31.

[2] Xu, X. W., and Q. He. "Striving for a total integration of CAD, CAPP, CAM and CNC." *Robotics and Computer-Integrated Manufacturing* 20.2 (2004): 101-109.

[3] Fletcher, Craig, et al. "The development of an integrated haptic VR machining environment for the automatic generation of process plans." *Computers in Industry* 64.8 (2013): 1045-1060.

[4] Wang, Hsu-Pin, and Jian-Kaing Li. "Computer-aided process planning."*Advances in Industrial Engineering* 13 (1993).

[5] Alting, Leo, and Hongchao Zhang. "Computer aided process planning: the state-of-the-art survey." *The International Journal of Production Research*27.4 (1989): 553-585.

[6] Ham, Inyong, and Stephen C-Y. Lu. "Computer-aided process planning: the present and the future." *CIRP Annals-Manufacturing Technology* 37.2 (1988): 591-601.

[7] Marri, H. B., A. Gunasekaran, and R. J. Grieve. "Computer-aided process planning: a state of art." *The International Journal of Advanced Manufacturing Technology* 14.4 (1998): 261-268.

[8] Weill, R., G. Spur, and W. Eversheim. "Survey of computer-aided process planning systems." *CIRP Annals-Manufacturing Technology* 31.2 (1982): 539-551.

[9] Cay, Faruk, and Constantin Chassapis. "An IT view on perspectives of computer aided process planning research." *Computers in Industry* 34.3 (1997): 307-337.

[10] Steudel, Harold J. "Computer-aided process planning: past, present and future." *THE INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH*22.2 (1984): 253-266.

[11] Chang, Ping-Teng, and Chia-Hua Chang. "An integrated artificial intelligent computer-aided process planning system." *International Journal of Computer Integrated Manufacturing* 13.6 (2000): 483-497.

[12] Lysenko, Mikola, Roshan D'Souza, and Keyvan Rahmani. "Real-time machinability analysis of free form surfaces on the GPU." *Journal of Computing and Information Science in Engineering* 9.2 (2009): 024504.

[13] Bi, Qing-Zhen, Yu-Han Wang, and Han Ding. "A GPU-based algorithm for generating collision-free and orientation-smooth five-axis finishing tool paths of a ball-end cutter." *International Journal of Production Research* 48.4 (2010): 1105-1124.

[14] Piegl, L., and W. Tiller. "The NURBS book, 1995."

[15] Shah, Jami J. *Parametric and feature-based CAD/CAM: concepts, techniques, and applications*. John Wiley & Sons, 1995.

[16] CAM-I (Computer Aided Manufacturing-International), 1989, Functional requirements for a feature based modeling system (CAM-I report)

[17] X. Xu*, X. W., et al. "STEP-compliant NC research: the search for intelligent CAD/CAPP/CAM/CNC integration." *International Journal of Production Research* 43.17 (2005): 3703-3743.

[18] Bi, Z. M., and W. J. Zhang. "Flexible fixture design and automation: review, issues and future directions." *International Journal of Production Research*39.13 (2001): 2867-2894.
[19] Kang, Xiumei, and Qingjin Peng. "Fixture feasibility: methods and techniques for fixture planning." *Computer-Aided Design and Applications* 5.1-4 (2008): 424-433.

[20] Hargrove, S. K., and A. Kusiak. "Computer-aided fixture design: a review."*THE INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH* 32.4 (1994): 733-753.

[21] Dong, X., W. R. DeVries, and M. J. Wozny. "Feature-based reasoning in fixture design." *CIRP Annals-Manufacturing Technology* 40.1 (1991): 111-114.

[22] Zhou, Yunbo, Yingguang Li, and Wei Wang. "A feature-based fixture design methodology for the manufacturing of aircraft structural parts." *Robotics and Computer-Integrated Manufacturing* 27.6 (2011): 986-993.

[23] Perremans, P. "Feature-based description of modular fixturing elements: the key to an expert system for the automatic design of the physical fixture."*Advances in Engineering Software* 25.1 (1996): 19-27.

[24] Dragomatz, Donald, and Stephen Mann. "A classified bibliography of literature on NC milling path generation." *Computer-Aided Design* 29.3 (1997): 239-247.

[25] Suh, Suk-Hwah, and Yang-Soo Shin. "Neural network modeling for tool path planning of the rough cut in complex pocket milling." *Journal of Manufacturing Systems* 15.5 (1996): 295-304.

[26] Held, Martin, Gábor Lukács, and László Andor. "Pocket machining based on contour-parallel tool paths generated by means of proximity maps." *Computer-Aided Design* 26.3 (1994): 189-203.

[27] Veeramani, Dharmaraj, and Yuh-Shying Gau. "Selection of an optimal set of cutting-tool sizes for 212D pocket machining." *Computer-Aided Design* 29.12 (1997): 869-877.

[28] Hinduja, S., et al. "Determination of optimum cutter diameter for machining 2 ½ pockets." *International journal of machine tools and manufacture* 41.5 (2001): 687-702.

[29] Liu, Zhenkai, and Lihui Wang. "Sequencing of interacting prismatic machining features for process planning." *Computers in Industry* 58.4 (2007): 295-303.

[30] Zhang, Hong-Chao, and Enhao Lin. "A hybrid-graph approach for automated setup planning in CAPP." *Robotics and Computer-Integrated Manufacturing*15.1 (1999): 89-100.

[31] Wu, H-C., and T-C. Chang. "Automated setup selection in feature-based process planning." *International Journal of Production Research* 36.3 (1998): 695-712.

[32] Ong, S. K., and A. Y. C. Nee. "Application of fuzzy set theory to setup planning." *CIRP Annals-Manufacturing Technology* 43.1 (1994): 137-144.

[33] Hebbal, S. S., and N. K. Mehta. "Setup planning for machining the features of prismatic parts." *International Journal of Production Research* 46.12 (2008): 3241-3257.

[34] Sakurai, Hiroshi. "Automatic setup planning and fixture design for machining."*Journal of Manufacturing Systems* 11.1 (1992): 30-37.

[35] Huang, Samuel H. "Automated setup planning for lathe machining." *Journal of Manufacturing Systems* 17.3 (1998): 196-208.

[36] Chen, Lin-Lin, and Tony C. Woo. "Computational geometry on the sphere with application to automated machining." *Journal of Mechanical Design* 114.2 (1992): 288-295.

[37] Sarma, Sanjay E., and Paul K. Wright. "Algorithms for the minimization of setups and tool changes in “simply fixturable” components in milling." *Journal of Manufacturing Systems* 15.2 (1996): 95-112.

[38] Kumar, A. Senthil, Andrew YC Nee, and S. Prombanpong. "Expert fixture-design system for an automated manufacturing environment." *Computer-Aided Design* 24.6 (1992): 316-326.

[39] Lee, Soo Hong, and M. R. Cutkosky. "Fixture planning with friction." *Journal of Manufacturing Science and Engineering* 113.3 (1991): 320-327.

[40] Lin, Rong-Shine. "Real-time surface interpolator for 3-D parametric surface machining on 3-axis machine tools." *International Journal of Machine Tools and Manufacture* 40.10 (2000): 1513-1526.

[41] Lee, Yuan-Shin. "Non-isoparametric tool path planning by machining strip evaluation for 5-axis sculptured surface machining." *Computer-Aided Design*30.7 (1998): 559-570.

[42] Chen, Y. D., J. Ni, and S. M. Wu. "Real-time CNC tool path generation for machining IGES surfaces." *Journal of Manufacturing Science and Engineering*115.4 (1993): 480-486.

[43] Elber, Gershon, and Elaine Cohen. "Tool path generation for freeform surface models." *Proceedings on the second ACM symposium on Solid modeling and applications*. ACM, 1993.

[44] Yang, Daniel CH, and Zhonglin Han. "Interference detection and optimal tool selection in 3-axis NC machining of free-form surfaces." *Computer-Aided Design*31.5 (1999): 303-315.

[45] Jensen, C. Greg, W. E. Red, and J. Pi. "Tool selection for five-axis curvature matched machining." *Computer-Aided Design* 34.3 (2002): 251-266.

[46] Chen, Zezhong C., and Qiang Fu. "An optimal approach to multiple tool selection and their numerical control path generation for aggressive rough machining of pockets with free-form boundaries." *Computer-Aided Design* 43.6 (2011): 651-663.

[47] Han, F. Y., et al. "Optimal CNC plunge cutter selection and tool path generation for multi-axis roughing free-form surface impeller channel." *The International Journal of Advanced Manufacturing Technology* 71.9-12 (2014): 1801-1810.

[48] Woo, Tony C. "Visibility maps and spherical algorithms." *Computer-Aided Design* 26.1 (1994): 6-16.

[49] Tang, Kai, J. Gan, and Tony Woo. Maximum intersection of spherical polygons and workpiece orientation for 4-and 5-axis machining. (1992)

[50] Attila, Rétfalvi. "IGES-based CAD model post processing module of a Setup and Fixture Planning System for box-shaped parts." *Intelligent Systems and Informatics (SISY), 2011 IEEE 9th International Symposium on*. IEEE, 2011.

[51] Wu, Y., et al. "Automated modular fixture planning: geometric analysis."*Robotics and Computer-Integrated Manufacturing* 14.1 (1998): 1-15.

[52] Brost, Randy C., and Kenneth Y. Goldberg. "A complete algorithm for synthesizing modular fixtures for polygonal parts." *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*. IEEE, 1994.

[53] Boonsuk, Wutthigrai, and Matthew C. Frank. "Automated fixture design for a rapid machining process." *Rapid prototyping journal* 15.2 (2009): 111-125.

[54] Bertoldi, M., et al. "Domain decomposition and space filling curves in toolpath planning and generation." *Proceedings of the 1998 Solid Freeform Fabrication Symposium, The University of Texas at Austin, Austin, Texas*. Academic Press, 1998.

[55] Yuwen, Sun, Guo Dongming, and Wang Haixia. "Iso-parametric tool path generation from triangular meshes for free-form surface machining." *The International Journal of Advanced Manufacturing Technology* 28.7-8 (2006): 721-726.

[56] Park, Sang C. "Sculptured surface machining using triangular mesh slicing."*Computer-Aided Design* 36.3 (2004): 279-288.

[57] Chen, Tao, and Zhiliang Shi. "A tool path generation strategy for three-axis ball-end milling of free-form surfaces." *Journal of materials processing technology* 208.1 (2008): 259-263.

[58] Lee, Sung-Gun, Hyun-Chul Kim, and Min-Yang Yang. "Mesh-based tool path generation for constant scallop-height machining." *The International Journal of Advanced Manufacturing Technology* 37.1-2 (2008): 15-22.

[59] Zhang, Ke, and Kai Tang. "An efficient greedy strategy for five-axis tool path generation on dense triangular mesh." *The International Journal of Advanced Manufacturing Technology* (2014): 1-12.

[60] Bi, Qing-Zhen, Yu-Han Wang, and Han Ding. "A GPU-based algorithm for generating collision-free and orientation-smooth five-axis finishing tool paths of a ball-end cutter." *International Journal of Production Research* 48.4 (2010): 1105-1124.

[61] Balasubramaniam, Mahadevan, et al. "Generating 5-axis NC roughing paths directly from a tessellated representation." *Computer-Aided Design* 32.4 (2000): 261-277.

[62] D'Souza, Roshan M., Carlo Sequin, and Paul K. Wright. "Automated tool sequence selection for 3-axis machining of free-form pockets." *Computer-Aided Design* 36.7 (2004): 595-605.

[63] Frank, Matthew C., Richard A. Wysk, and Sanjay B. Joshi. "Determining setup orientations from the visibility of slice geometry for rapid computer numerically controlled machining." *Journal of manufacturing science and engineering* 128.1 (2006): 228-238.

[64] Li, Y. and Frank, M.C., Computing Axes of Rotation for Setup Planning Using Visibility of Polyhedral CAD Models, *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, 2012,134(4), 041005(1-10).

[65] Li, Ye, and Matthew C. Frank. "Machinability analysis for 3-axis flat end milling." *Journal of manufacturing science and engineering* 128.2 (2006): 454-464.

[66] Li, Ye, and Matthew C. Frank. "Computing non-visibility of convex polygonal facets on the surface of a polyhedral CAD model." *Computer-Aided Design* 39.9 (2007): 732-744.

[67] Haghpassand, K., and James H. Oliver. "Computational geometry for optimal workpiece orientation." *Journal of Mechanical Design* 117.2A (1995): 329-335.

[68] Dhaliwal, Savinder, et al. "Algorithms for computing global accessibility cones."*Journal of Computing and Information Science in Engineering* 3.3 (2003): 200-209.

[69] Spyridi, Antonia J., and Aristides AG Requicha. "Accessibility analysis for polyhedral objects." *Engineering systems with intelligence*. Springer Netherlands, 1991. 317-324.

[70] Vafaeesefat, A., and H. A. ElMaraghy. "Optimal workpiece orientations for machining of sculptured surfaces." *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 214.8 (2000): 671-681.

[71] Lasemi, Ali, Deyi Xue, and Peihua Gu. "Recent development in CNC machining of freeform surfaces: A state-of-the-art review." *Computer-Aided Design* 42.7 (2010): 641-654.

CHAPTER 3.AUTOMATED SETUP PLANNING FOR FEATURE FREE MULTIPLE SURFACE PARTS

Ashish M. Joshi, MS, and Matthew C. Frank, PhD

Department of Industrial and Manufacturing Systems Engineering,

Iowa State University, Ames, IA 50010, USA

Donald D. Anderson, PhD, Thaddeus P. Thomas, PhD, M. James Rudert, PhD,

And Thomas D. Brown, PhD

Department of Orthopedics and Rehabilitation

The University of Iowa, Iowa City, IA 52242, USA

Abstract

This paper presents a new computer aided process planning (CAPP) algorithm for the rapid CNC machining of feature free multi-surface parts (MSP), defined as parts with geometry having multiple unique surface characteristics such as roughness, texture, hardness, or various other mechanical surface properties. This work considers multiple sub-surfaces and presents setup planning algorithms for their rapid machining. A new algorithm written in C++ using constrained combinatorial optimization and the meta-heuristic simulated annealing (SA) is introduced for determining setup orientations in automated machining. The setup orientations facilitate increased line of sight visibility and depth-wise reachability of an isolated surface area for a machine tool, in the midst of other surfaces on a given MSP. This capability enables the use of surface-specific machining parameters and the creation of MSPs with varying surface textures. Introducing rapid machining for fabricating MSPs could have a major impact in terms of increased ability to machine more feature-free and complex parts with multiple sub-surface properties and with improved functionality of MSPs. Potential applications from

the algorithm in this work could be used for finish machining of selective critical surfaces on castings, which would allow for the creation of low tolerance surfaces. Additionally this method could also be used for finish machining of additively manufactured (AM) near net-shape components, which require some level of post-process machining, for example, targeting of critical surfaces on metal powder based RP components. In this paper, this process is shown as applied to bone implants, notably for patient-specific implants for orthopedic trauma management, which provides a means to potentially enhance the primary (mechanical fixation) and secondary (biologic) stability of the implants, to improve fracture treatment, and to reduce treatment time and cost.

Keywords – Rapid Machining, Setup Planning, Feature Free, Multiple Surface Part (MSP), Bone Implant, Surface Texturing

3.1 INTRODUCTION

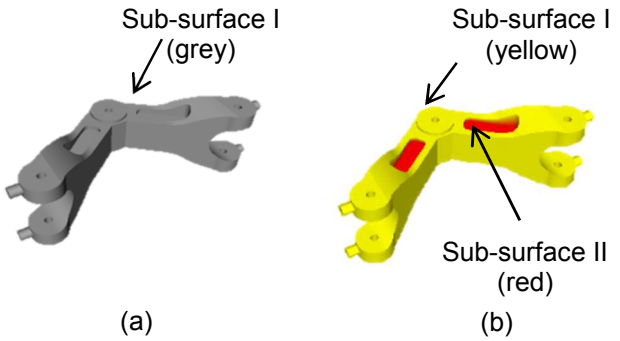


Figure 1: Prismatic components (industrial parts) (a) Multi-surface part (b) Single surface part

Computer aided process planning (CAPP) for manufacturing has been an invaluable tool to manufacture parts using automated additive manufacturing (AM) processes using polygonal de-facto STL models. This was extended to the subtractive rapid manufacturing (SRM) process of rapid CNC machining when automated setup planning algorithms were first developed by Frank et al [1] for determining machining setup orientations. These algorithms enabled rapid machining of

industrial and biomedical parts using automatically determined machining orientations. However, with the advent of the 21st century, there is a growing need for developing advanced rapid manufacturing processes that can make parts with customized characteristics. These customized characteristics could include a broad variety of volumetric or surface attributes such as density variation, material combinations, ductility, and malleability, material coating, texture, hardness, toughness, and roughness, etc. in order to satisfy the functional intent. This paper presents new CAPP algorithms for rapid CNC machining of a class of parts termed as multiple surface parts (MSP). One application of these parts, manufactured through rapid machining, is industrial and biomedical applications for advanced functional intent.

3.1.1 Single surface vs. Multiple Surface Parts (MSPs)

Single surface parts have same-of-a-kind property over their entire surface geometry. For example, a complete as-cast part requiring no post processing on it or a part machined from a stock with equal tolerances over the entire geometry (Figure 1.a) is a single surface part. By comparison, multiple surface parts (MSPs) may be characterized by gradual variation in characteristics throughout the surface geometry, resulting in corresponding changes in the material or mechanical properties such as surface roughness values, hardness, texture types, color, etc. This would include part geometry that can be identified as a collection of multiple surfaces that either need fabrication or post processing using different methods. For example, a machined casting could be considered an MSP, which would have a cluster of surfaces (Figure 1.b) machined to hold critical tolerances as compared to other surfaces which are left untouched and are acceptable as as-cast surfaces.

3.1.2 Bone implants as Multiple Surface Parts (MSPs)

One challenging example of freeform MSPs includes bone implants, where each of a bone's surfaces has specific physical characteristics to enhance biologic and mechanical functionality (Figure 2). Segmental bone defects pose major challenges for orthopedic management. Whole sections of bone are oftentimes missing or must be surgically removed during the treatment of disease or trauma. For bone to function properly, it is essential that a defect be filled with an implant that is both mechanically and biologically stable. These characteristics are affected by geometry as well as roughness. Shaping an implant from clinically relevant material can be challenging; currently, surgeons often sculpt these implants by hand during surgery to fit the defects. Because bones have multiple surface types, this task is prone to inaccuracies which can lead to complications, particularly for joint fractures, where a poorly filled bone defect can alter joint mechanics, compromise an implant's primary fixation stability, and ultimately cause a joint to degenerate. Besides creating a bone implant with accurate geometry, the ability to generate specific textures and roughness on different surfaces of an implant is also critical for increasing an implant's primary fixation stability which is necessary for effective healing.

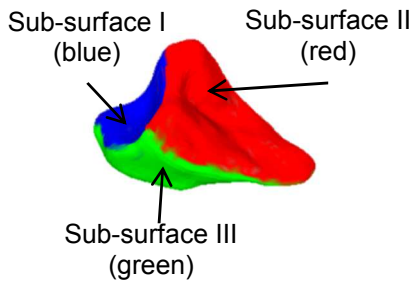


Figure 2: Bone implant as MSP

3.1.3 Manufacturing using rapid CNC machining

The rapid CNC machining process developed at the Rapid Manufacturing & Prototyping Lab (RMPL) at Iowa State University is a fully functional rapid manufacturing process and is abbreviated as RM process throughout this paper. The RM process uses a standard 3-axis CNC milling machine with a 4th axis indexer for multiple setup orientations. This machining process includes completely automated fixture planning, tooling, and setup planning, including generation of NC code for creating a part directly from feature free CAD models (Frank et al. [1][2][3][4][5]). The use of a rotation axis eliminates the need for re-clamping of the part, a common task in conventional fixturing methods (Li and Frank, 2006, 2007). For each orientation, all of the visible surfaces are machined while a set of sacrificial supports keeps the part connected to the uncut end of the stock material. Once all of the operations are complete, the supports are severed (sawed or milled) in a final post processing step and the part is removed [3], the setup and steps to this process are illustrated in Figure 3. The manufacturing of biomedical implants provides a challenge well-suited for the RM process, especially because of the fixturing issues and the need for specialty materials (in particular, human allograft bone).

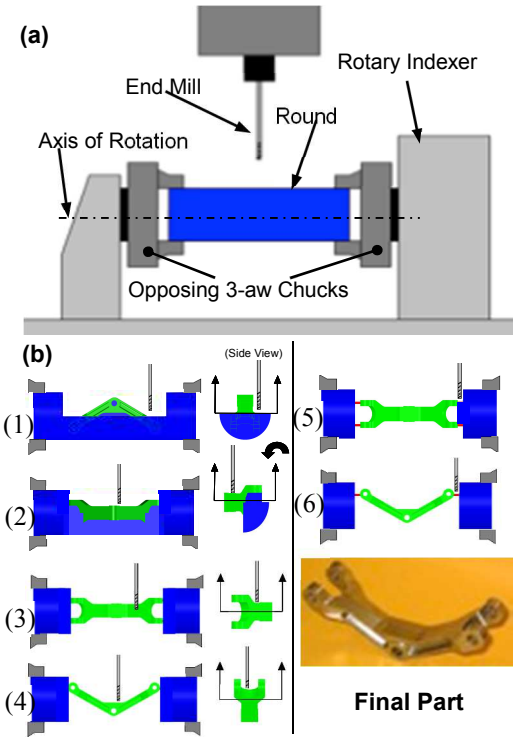


Figure 3 - (a) RM setup; (b) Steps b.1-b.4 expose component geometry while b.5-b.6 expose sacrificial supports

3.1.4 Surface texture

In its new instantiation, the RM process has the capability to customize the roughness for different surface types. The required texture and characteristics of multiple different surfaces can be imparted through machining by using specifically planned tool paths on each surface, avoiding the tedious task of designing textures on feature free CAD models.

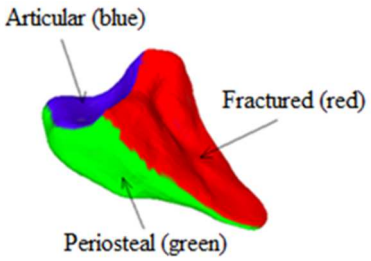
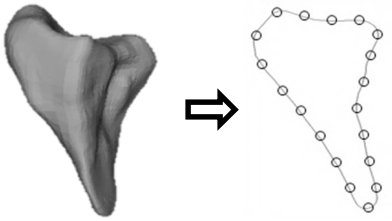


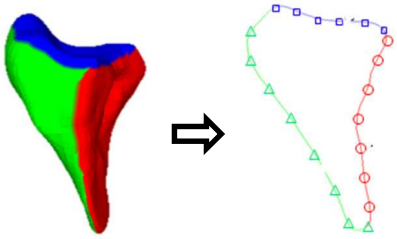
Figure 4: PLY format: implant sub-

For the case of segmental bone defects that extend into the joint, it is desired that an implant which will fill a defect have three distinct surface types, analogous to those of bone fragments: *articular*, *periosteal,* and *fracture* (Figure 4). The *articular* surface is that which would be in contact with apposing cartilage in a moving joint; the *periosteal* surface is that which would be in contact with other soft-tissues (e.g., muscles), while the fracture surface would contact the surface created during the fracture event. The articular surface always has the lowest roughness (smooth: type1) compared to the other two surfaces. The periosteal surface requires only a medium roughness (medium: type2), and the fracture surface generally has high roughness (rough: type3), since it breaks off from the parent bone during the fracture event.



(a) (b)

Figure 5 a) Uncolored STL file b) 2D uncolored STL slice



(a) (b)

Figure 6 a) Colored PLY file b) 2D colored PLY slice

In spite of significant progress in the setup planning methods, the use of feature free models for addressing this problem is minimal to non-existent. There is an essential need for addressing setup planning problems that can be solved using feature free models. The work presented in this paper hence aims to provide a robust method for automated setup planning using feature free CAD models for machining parts with customized surface characteristics, and the use of RM to generate viable multi-textured bone implants demonstrates the efficacy and efficiency of this work.

3.1.5 Problem statement

Previous algorithms for RM have determined setup orientations using feature free models which focus on generating a model's entire geometry, but those algorithms have not generated different types of targeted surfaces in the models. Hence the objective in this paper is to choose a set of orientations for multiple surfaces: in this example three bone surfaces are created, such that type1 roughness is created on articular surfaces, type2 roughness is created on periosteal surfaces, and type3 roughness is created on fracture surfaces. To do this, a basic *Set Cover* approach can be used, but an individual cover for each targeted surface is generated rather than one single set cover solution for the entire model. The problem is, then, how to target individual surfaces choosing the minimal number of setup orientations aimed per surface as part of the entire model geometry. This requires the use of what will be deemed a constrained set cover approach.

We present an accurate, highly automated, and efficient method for creating defect-specific multiple surface parts with multiple custom textures using an automated rapid CNC machining process. We demonstrate this in a biomedical application to advance

the surgical treatment of debilitating segmental bone defects. A set cover with constrained combinatorial optimization using the heuristic simulated annealing (SA) is employed to determine the setups required to achieve multiple custom textures. The CAD geometries of segmental defects (residual voids) were extracted and designed using fracture reconstruction planning software called (Thomas et al [6]) to analyze patient CT data. The corresponding custom defect-specific implant specimens were then fabricated from cylindrical stocks of 40 lbs/ft^3 $BaSO_4$ infused polyurethane using rapid machining on a 4-axis CNC mill. The primary attraction of rapid CNC machining in this application is the ability to closely match defect geometry and surface finishes. In the example application, the ability to do so provides improved primary fixation stability of a bone/implant construct which advances the treatment of segmental bone defects.

3.2 Methods

3.2.1 STL vs. PLY files

The file format used for process planning was PLY format (Figure 6a), instead of the de-facto standard STL file that is used in conventional rapid manufacturing processes. Similar to the STL format (Figure 5a), the PLY file format uses triangular facets for part geometry approximation. PLY also allows for efficient storage of a variety of surface properties: color, transparency, surface normal, etc. for each facet. Thus, the color stored on a specific group of facets acts like a surface identifier. Further, for setup calculations, PLY files are sliced similarly to STL files, orthogonal to the chosen rotary axis. Each slice consists of multiple simple polygons (chains) represented by the end-points of the polygon segments (edges of the polygon) (Figure 6b). In the example presented for distinguishing surfaces from one another, the points on the 2D segments

on articular/periosteal/fracture surface chains are represented using the colors blue, green, and red, as well as the symbols (▫,△,○), respectively (Figure 6b).

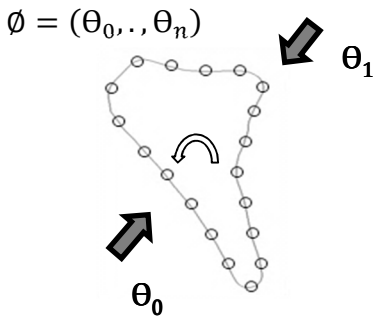


Figure 7: Setup orientations using STL slices

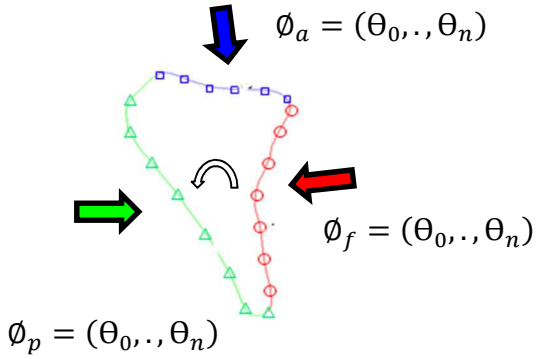


Figure 8:Setup orientations targeting individual surfaces using PLY slices

Followed by PLY slicing, the visibility-based setup orientations are determined using a set-covering constrained combinatorial optimization and the simulated annealing heuristic. The objective function designed for evaluating a set of surface specific orientations represents its goodness score, which determines the set of orientations to be chosen for machining the implant surfaces. As in the RM process, 2½ tool paths for machining a model surface portion are executed from each prescribed setup orientation. However, the PLY file format allows further custom finishing operations for each surface type, once setup orientations are isolated to individually cover (i.e., machine) each surface.

3.3 Process Planning For Calculating Surface Specific Orientations

As mentioned previously, the problem of determining the set of setup orientations for machining an entire part is classified as a set covering problem, where the entire surface of a part model visible in the range of 0° to 360° is included in the universal set (Figure 7). In previous work for RM process planning, it was deemed necessary that the whole surface of a part model be machined only after all setup orientations were

completed. Due to the lack of surface identification on the STL file, the previous algorithms for determining orientations were designed to target the entire model geometry, but did not create different characteristics on each surface.

The algorithms designed for the RM process ensure that every region on the part surface visible in the range of 0° to 360° is machined from at least one setup orientation from the chosen solution set. Using the mapped visibility ranges for each segment on the slice file, the minimum number of setup orientations required for machining a part are calculated (Figure 7). With colored PLY models, planning for setup orientations that are aimed at specific surfaces and that allow specific characteristics while avoiding machining other surfaces are allowed. The basic set cover approach is used here, but constraints are applied in order to achieve more targeted covering for each surface individually, rather than for the entire model arbitrarily. Thus, in order to target individual surfaces, setup orientations must be chosen such that they are aimed at surfaces individually (Figure 8) rather than at multiple surfaces together. By its very nature, this heuristic approach may or may not achieve optimal results, since the set of minimum setups in an unconstrained case would often be fewer.

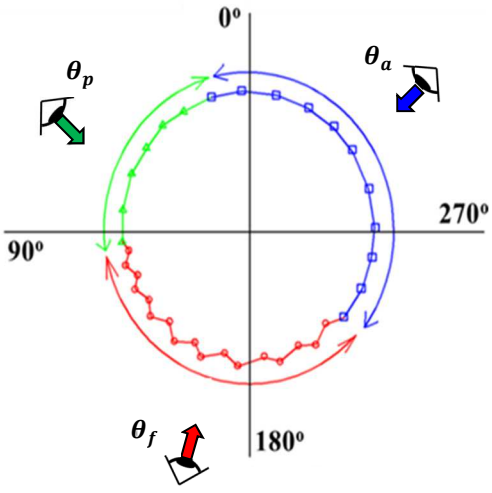


Figure 9 : Surface Visibility

In the example shown in this work, setup orientations specific to the articular/periosteal/fracture bone surfaces are designated with subscript $\theta_{a/p/f}$. The process planning algorithm developed for choosing surface specific setup orientations considers

the primary variables of 1) surface visibility (SV), 2) surface reachability (SR), and 3) normal deviation (ΔN) (deviation of a candidate setup orientation away from the average surface normal).

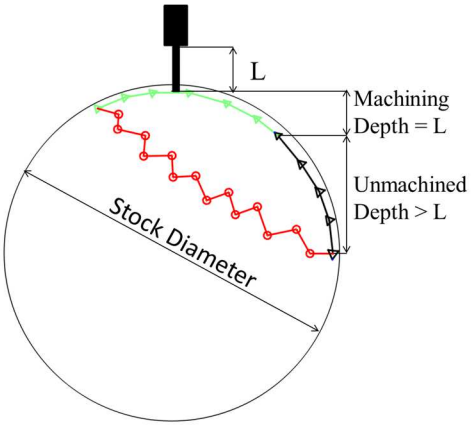


Figure 10 : Surface Reachability, tool Length L < Depth D, inaccessible

3.3.1 Surface Visibility (SV)

Visibility is a necessary condition for a given surface to be machined by CNC-RP, and is given by [A]: $SV_{i, j, k, p/a/f} \neq [\Phi]$ (null set), for all segments (i), on all chains (j), of all slices (k) for *p*, *a* or *f* surface.

For example, if the objective is to machine the articular surface, a candidate set of setup orientations is one from which the complete articular surface is visible (Figure 9). The primary aim is then to maximize the visibility of the surface for the chosen set of setup orientation(s).

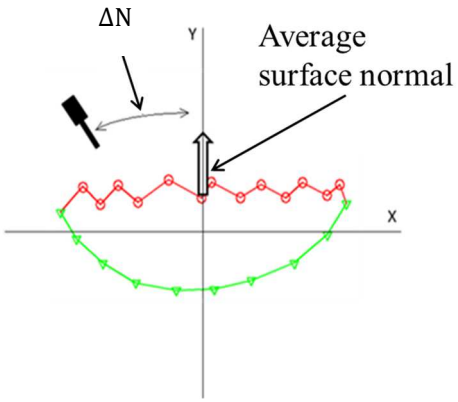


Figure 11: Normal Deviation difference between orientation and average surface normal

3.3.2 Surface Reachability (SR)

In addition to surface visibility, surface reachability is a sufficient condition for machining the visible surface and is given by:

$SR_{i, j, k, p/a/f} \neq [\Phi]$ (null set), for all segments (i), on all chains (j), of all slices (k) for *p*, *a* or *f* surface.

To ensure that an entire part can be machined, it is required that every visible surface on the part also be reachable from at least one candidate orientation. For example,

there could be an instance where a certain percentage of a surface is visible from a given orientation, but is not reachable because of inadequate tool length. In this example, it can be seen in Figure 10 that the periosteal surface being machined by a tool with length L cannot access certain sites whose depth values are greater than L. Thus, the total reachable perimeter is calculated by comparing the maximum available tool length against the perpendicular distance from each visible point to the tangent line at the given orientation (Figure 10). Hence, while choosing setup orientations, the aim is to minimize the surface reachability for unintentional surfaces using a tool of length L but that will completely machine the surface of primary interest. To ensure complete accessibility to the reachable surface of primary interest, it is assumed that a proper tool is chosen with a sufficiently small radius.

3.3.3 Normal deviation (ΔN)

Texture directionality and magnitude created on a surface is also a necessary factor to consider. For a bone implant to have primary fixation stability, it is desirable that the quasi-pyramidal texture (Figure 4) be created normal to the fracture surface, increasing implant stability and allowing efficient and faster bone in-growth (Table 1). It is also desirable to have minimal ΔN to maintain type 2 & 1 roughness on the periosteal and articular surfaces (Table 1). Hence, to maintain the required directionality of the created texture and the implant's biomechanical compatibility, it is desirable to have the tool axis inclination (or the Normal Deviation, ΔN) as low as possible. The overarching goal is to have *minimal* ΔN (Figure 11) when choosing setup orientations.

$$\text{Normal Deviation } (\Delta\text{N}) = \vartheta(fabs\,((\theta_i - \theta_n)/90.0))$$

Table 1: Influences on surfaces due to variation in Normal Deviation

Surfaces	Normal Deviation	
	High	Low
Fracture	Reduces fixation stability	Increases fixation stability
Periosteal	Soft tissue irritation	Maintains biomechanical compatibility
Articular	Cartilage wear	Maintains biomechanical compatibility

3.3.4 Goodness measure for a set of setup orientation for a surface of primary interest

For a given candidate set of setup orientations for a specific surface, a quantitative goodness measure is defined by four customization control parameters: 1) tool path containment (*TCO*); 2) tool path crossover (*TCR*); 3) tool path redundancy (*TR*), and 4) normal deviation(ΔN). While normal deviation is the difference between the candidate orientation and the average normal for the surface of primary interest, the other three parameters are functions of surface visibility (*SV*) and surface reachability (*SR*). A description of each is given in the following sections, where the customization control parameters of a surface of primary interest is designated with subscript "*s,*" while unintentional surfaces are designated with subscript "*us.*"

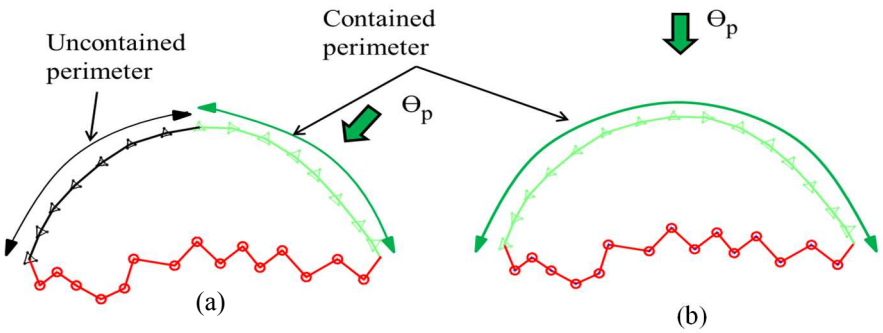


Figure 12: Tool path Containment a) Partial Surface contained by an orientation b) Complete Surface contained by an orientation

3.3.5 Tool Path Containment (*TCO*)

Tool path containment (TCO) is the idea of machining only the visible and reachable surface of primary interest. Since the overall goal is to choose the minimum number of setup orientations to machine a surface, a set of setup orientations with a maximum percentage of TCO will always be chosen to machine the maximum surface. For example, for machining a bone fracture surface (Figure 12) the setup orientation with the maximum percentage of TCO of the fracture surface will be chosen.

$$TCO_s = \left\{ \beta \left(\sum_{\theta=0}^{z} \left\{ \sum_{i=0}^{l} \sum_{j=0}^{m} \sum_{k=0}^{n} \{(SV)_{i,j,k}\} \right\}_{\theta} \Big/ SV_{total} \right) + \sum_{\theta=0}^{z} \alpha L_{\theta} \right\}_s$$

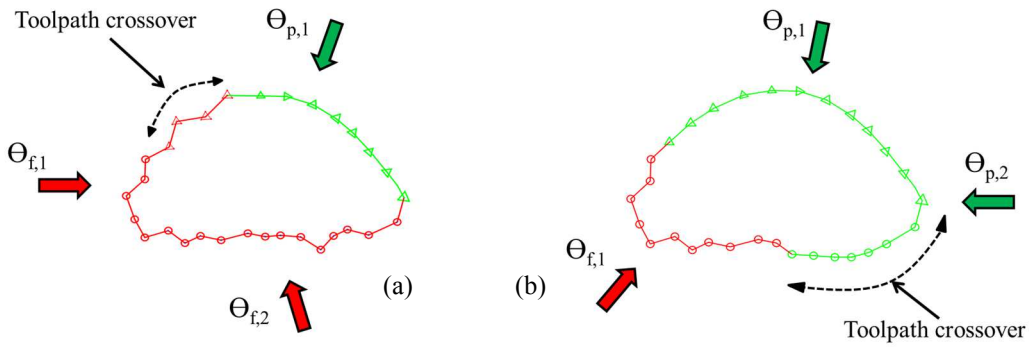


Figure 13: Illustration of tool path crossover (a) tool path crossover from fracture

3.3.6 Tool Path Crossover (*TCR*)

Tool path crossover (*TCR*) occurs if the machining is performed using a set of orientations from which other surfaces that are unintentionally visible are covered *in addition to* the surface of primary interest (Figure 13). *TCR* can wipe out an intended generated texture. For example, while choosing orientations for the periosteal surface on a bone, the "*TCR* to fracture surface" means that the fracture surface would also be machined when the intention was to machine only the periosteal surface, potentially reducing the primary fixation stability of the implant. Because of the practical

effectiveness of customized implant, the goal is simple: machine surfaces of primary interest with customized tool paths, and avoid machining unintentional surfaces while doing so. Hence, the setup orientations giving *minimal TCR* are chosen for machining. Table 2 shows the effect of *TCR* on unintentional surfaces.

$$TCR_{us} = \sum_{us=0}^{y} \left\{ \gamma \left({\sum_{\theta=0}^{z} \left\{ \sum_{i=0}^{l} \sum_{j=0}^{m} \sum_{k=0}^{n} \left\{ (SV)_{i,j,k} \right\} \right\}_{\theta}} \Big/ {SV_{total}} \right) + \delta \left(\sum_{\theta=0}^{z} \left\{ \sum_{i=0}^{l} \sum_{j=0}^{m} \sum_{k=0}^{n} \left\{ {\{(SR)_{i,j,k}\}} \Big/ {\{(SV)_{i,j,k}\}} \right\} \right\}_{\theta} \right)_{L} \right\}_{us}$$

Table 2: Effects on different surfaces due to tool path crossover

Tool path requirements	Surface to Surface	Effect
Tool path Crossover	Articular to Periosteal	Inefficient machining
	Articular to Fracture	Reduced fixation stability
	Periosteal to Articular	Cartilage wear (moderate)
	Periosteal to Fracture	Reduced fixation stability
	Fracture to Articular	Cartilage wear (severe)
	Fracture to Periosteal	Soft tissue irritation

3.3.7 Tool Path Redundancy (*TR*)

Tool path redundancy (TR) is unessential or redundant machining of a common perimeter reachable from multiple orientations for the surface of primary interest (Figure 14). In the case of bone implants, redundant machining on a periosteal or articular surface would be inefficient since additional smoothing of those surfaces is unnecessary. Unnecessary increased machining time could negatively impact the practical use of this technology in a production setting (cost, machine capacity, etc.). Furthermore, redundant machining of a fracture surface could also reduce texturing effects (or ablate them completely), once again having a real effect by potentially

leading to a reduction in primary fixation stability of the implant. Hence, the setup orientations giving *minimal TR* are chosen for machining.

$$TR_s = \varepsilon\left(\left(\frac{\sum_{\theta=0}^{z}\{\sum_{i=0}^{l}\sum_{j=0}^{m}\sum_{k=0}^{n}\{(SR)_{i,j,k}\}\}_{\theta}}{SR_{total}}\right)_s - 1.0\right)$$

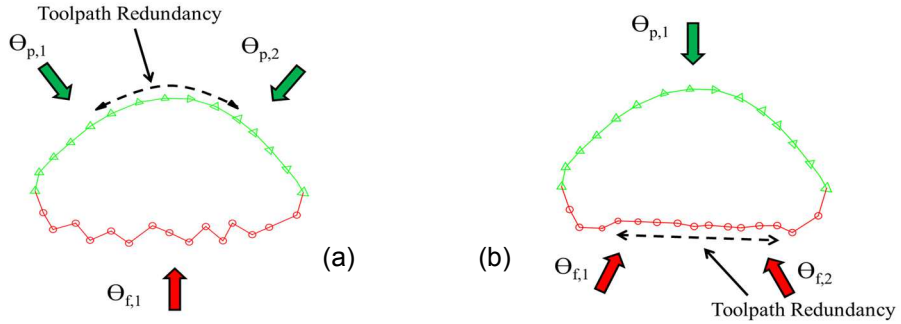


Figure 14: Toolpath redundancy in the (a) periosteal or (b) fractured sub-surfaces

Table 3: Effects due to tool path redundancy in the case of bone implants

Tool path requirements	Surface to Surface	Effects
Tool path redundancy	Articular	*Wasted machining resources*
	Periosteal	*Wasted machining resources*
	Fracture	*Reduction in fixation stability*

3.3.8 Multi-objective function using greedy heuristic

A multi-objective function is proposed that aids in choosing a set of orientations for each surface "*s*" of primary interest considering unintentional surfaces "*us*" that a) *maximize* toolpath containment (*Max* TCO) b) *minimize* tool path crossover on unintentional surfaces, (*Min* TCR) c) *minimize* tool path redundancy, (*Min* TCR) and 4) *minimize* the normal deviation for a targeted sub-surface, (*Min* ΔN). The objective function formulated as a minimization problem is as follows:

$$\emptyset_s = (\Theta_0, \Theta_1, .., \Theta_n)_s$$

$$Find\ \emptyset_s = (\Theta_0, \Theta_1, .., \Theta_n)_s$$

$$Min(f(\emptyset_s)) = Min\{[\mathrm{TCR}]_{us} + [\mathrm{TR}]_s + [\Delta \mathrm{N}]_s - [\mathrm{TCO}]_s\}$$

3.3.9 Stochastic combinatorial optimization using Simulated Annealing (SA)

The problem of determining setup orientations is a set cover problem and falls under the category of NP-Hard problems. NP-Hard problems have no known polynomial time algorithms for solving them. In this paper, therefore, a stochastic combinatorial optimization approach using simulated annealing (SA) was employed. Simulated annealing is a probabilistic meta-heuristic algorithm used for combinatorial optimization using an objective function for evaluating goodness of a candidate solution. Physically, annealing is a process of cooling down a metallic object heated past its melting point. The rate of cooling decides the quality of crystalline structure formed during the process. An optimally controlled cooling rate can produce crystals of the necessary size while a faster rate can cause premature and imperfect crystallization. SA starts at a higher temperature us sub-optimal solution and optimizes on it further, until either the global optimum or an acceptance criterion is satisfied.

3.4 Determining Surface Specific Setup Orientations Using SA

Similarly, a feasible solution for a given surface is a set of orientations that will allow machining with sub-optimal customization. A feasible solution was obtained using a greedy heuristic that was optimized further using simulated annealing. A greedy algorithm was applied about the chosen rotation axis on the PLY slice model and a series of feasible solutions for each surface was evaluated. The best greedy solution from each surface was selected and SA was employed to improve the surface qualities using the objective function and probabilistic approach. Accepting probability is the

probability that the objective function accepts inferior solutions. At higher temperatures, the algorithm is dynamic in nature, which results in a high probability of jumping out from locally optimal solutions, allowing for the exploration of better solutions. The acceptance probability given by a Metropolis criterion is a function of the system temperature (T) and the behavior of the objective function (ΔE). As the temperature of the system decreases, the probability of accepting an inferior move is decreased. This is the same as gradually moving towards a fixed state in the actual annealing process. Similarly, if the temperature is reduced down to zero then only superior moves are accepted.

The presented method including the customization control parameters, the designed objective function, a greedy heuristic for obtaining a seed solution, and the subsequent simulated annealing procedure were implemented. Figures 15.a shows a bone implant with three and two sub-surfaces present on them respectively, chosen for determining the machining orientations for the RM process. The SA procedure was carried out with

Table 4: Simulated Annealing results on different sub-surfaces

Results	Articular (Blue)	Periosteal (Green)	Fractured (Red)
Orientation	199	275	84
Seed orientation	10	180	0
Iterations	104	110	227
Tool path containment	1.0	1.0	1.0
Tool path crossover	0.0	0.891	0.928
Tool path redundancy	-	-	-
Average normal deviation	0.132	0.348	0.198

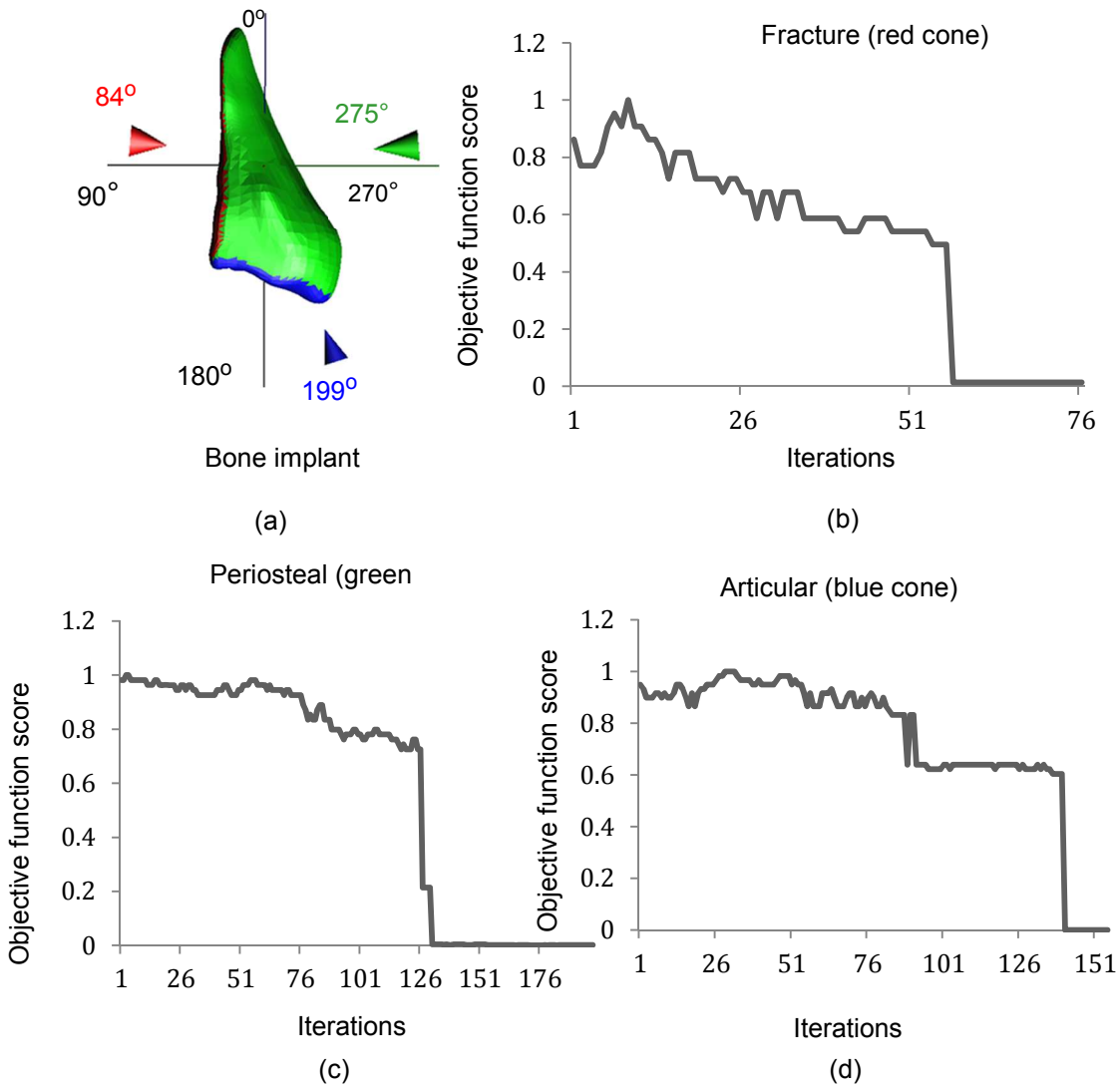


Figure 15: a) Sub-surface wise setup orientations b) Convergence fractured sub-surface c) Convergence periosteal sub-surface d) Convergence articular sub-surface

various candidate annealing schedules and the graphs in Figures 15 show the convergence process for a suitable chosen schedule for each sub-surface type on an implants for a given seed solution. In this paper, guidelines from the Kirkpatrick et al. were closely followed in order to use a suitable annealing scheme. The initial temperature, Tinitial, was set according to equation Tinitial = Δ/Pinitial, where Δ is the average increase in the objective function score. The initial acceptance probability, Pinitial, was decided considering a fraction of uphill transitions in a trail run. The

decision for choosing Epoch length, i.e., the number of iterations per temperature level, was based on consideration of the number of surfaces present on the implant and their visibility ranges about a 0^o-360^o basis used by the RM rotary axis. This prevented redundant evaluation of design space spanning 0^o-360^o. The cooling schedule was chosen based on various trial and error runs for different surfaces on different implants, and the temperature reduction factor was chosen to as approximately 0.1-0.25 per level.

To test the reliability of the chosen SA schedule towards the effectiveness of creating customized sub-surface geometries, twenty repetitive trials were conducted on each of the sub-surfaces present on the implants and the corresponding setup and customization results were obtained. The results show a robust overall convergence towards setup solutions that would provide customized sub-surfaces allowing better and improved functionality of the machined implant.

Table 5: Customization results on different sub-surfaces

Average results	Articular (Blue)	Periosteal (Green)	Fractured (Red)
Iterations	134	94	184
Tool path containment	1.0	1.0	1.0
Tool path crossover	0.0	0.826	0.87
Tool path redundancy	-	-	-
Average normal deviation	0.25	0.29	0.195

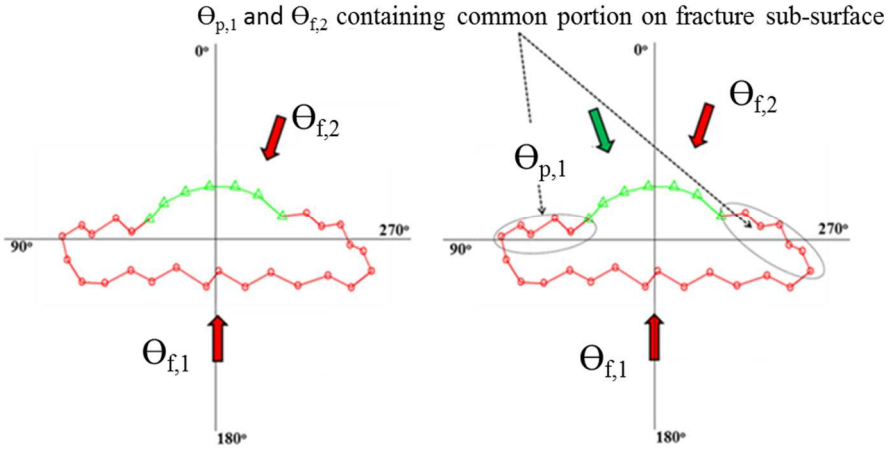


Figure 16: a) θ_{f1} and θ_{f2} calculated first b) θ_{p1} calculated later

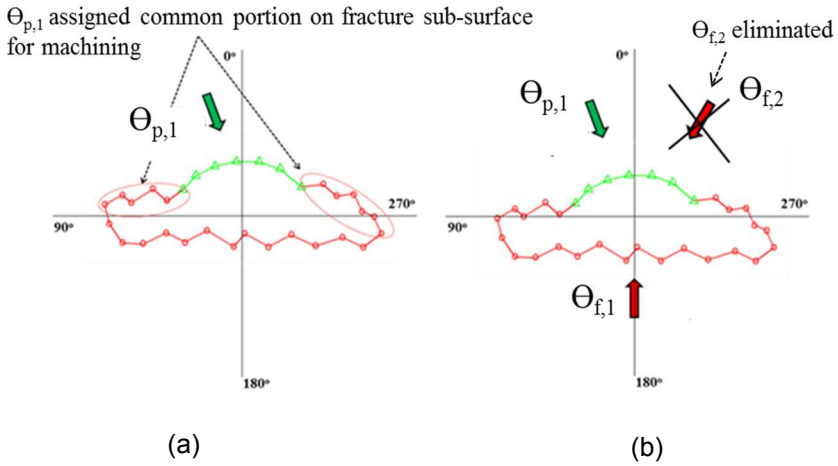


Figure 17: a) θ_{p1} calculated first b) Redundant orientation θ_{f2} eliminated

3.5 Setup Orientation Calculation Sequence

Using a surface specific sequence for determining machining orientations can reduce the amount of setup orientations required to machine the entire model. Figure

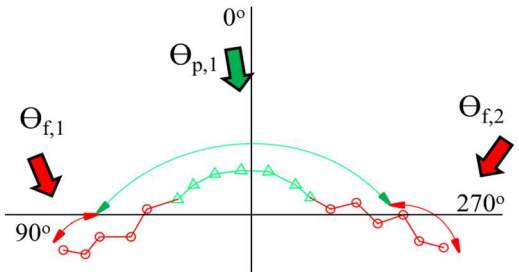


Figure 18: Θ_{p1}, Θ_{f1}, and Θ_{f2} required for machining periosteal and fracture surfaces

16.a shows for a bone example a chain with a relatively small periosteal surface and a larger fracture surface. It can be seen that there is a need for at least two orientations for machining the fracture surface, while for the periosteal surface only one setup orientation is necessary. The orientation Θ_{f1} will machine the fracture surface contained in the range of 90° to 270°, while the orientation Θ_{f2} will create the rest of the fracture surface contained within the 270° to 90° range. Θ_{f2} will also create the rough texture on the periosteal surface. The orientation Θ_{p1} will create the smooth finish on the periosteal surface contained within the 270° to 90° range; however, it will also destroy a portion of the rough texture created on the fracture surface created by Θ_{f2} (Figure 19b). This shows that "*destructive interference*" due to Θ_{p1} on the fracture surface is inevitable due to the containment of both surfaces in a common range (270° to 90°). For the bone example, in order to maintain biocompatibility, since it is acceptable to have a smooth texture on a rough surface, although not the converse, the use of the Θ_{f2} orientation would be redundant. This can be eliminated if surfaces for

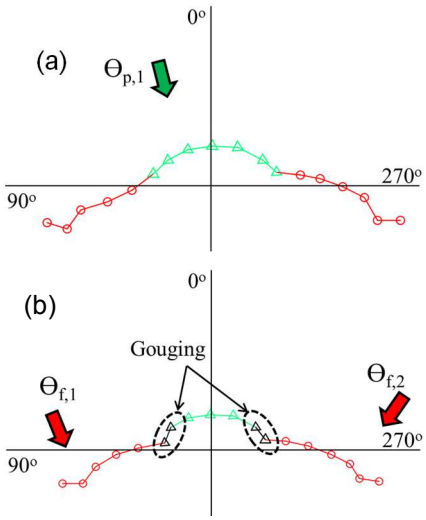


Figure 19: a) Θ_{p1} creating periosteal and fracture surface perimeter b) Θ_{f1} and Θ_{f2} gouging into periosteal surface

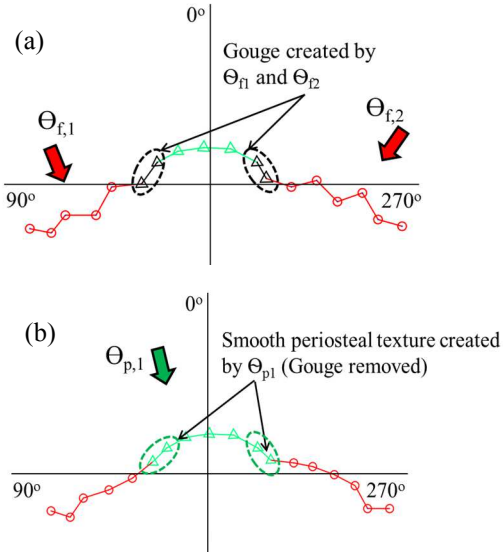


Figure 20: a) Θ_{f1} and Θ_{f2} create fracture surface first b) Θ_{p1} creates smooth

which orientations are to be determined are considered in a specific sequence. Hence, for the above case, if the orientations for the periosteal surface are calculated first, the fracture surface perimeter visible from those orientations is excluded when determining orientations for the fracture surface (Figure 17a). This avoids the use of redundant orientations for both surfaces present on the implant geometry (Figure 17b). It is advisable to determine the orientations for the smoothest surfaces first, followed by the medium rough surfaces, finally the roughest surfaces for the case of part like a bone implant having all three surfaces on it.

3.6 Machining Sequence

In addition to the sequence in which surfaces are considered for determining the setup orientations, the sequence in which surfaces are machined is also important. For example when considering the complex geometry of a bone defect implant, there is often a unique fracture surface orientation required which may unintentionally gouge into periosteal or articular surfaces. Unintentional machining or gouging could affect an implant's physiologic suitability. It would be unacceptable, for example, to incur a rough texture or a gouge on either the periosteal or articular surface. Therefore, it would always be better to machine the roughest (in this case fracture) surface first, followed by the periosteal and then the articular surface. The idea is to allow a smoother finish on a surface which requires rougher texture, rather than conversely, by compensating for Tool Path Crossover. Figure 18 illustrates a case in which the periosteal and fracture surfaces are present on the chain. Orientation Θ_{p1} is necessary to create a smooth periosteal surface while orientations Θ_{f1} and Θ_{f2} are necessary to create the rough texture on the fracture surface. If the machining sequence used in this case is Θ_{p1} and

then Θ_{f1}, Θ_{f2} (Figure 20), a smooth texture would be created on the contained periosteal and some of the fracture surface by $\Theta_{p1,}$ followed by the creation of rough texture on fracture surface by Θ_{f1} and Θ_{f2}. Potentially the tool paths from orientations Θ_{f1} and $\Theta_{f,2}$ could gouge into the periosteal surface. But, if the fracture surface on the implant is machined first using Θ_{f1} and $\Theta_{f,2}$, the gouge created on the periosteal surface resulting from these orientations would be replaced with the smooth finish using orientation Θ_{p1} (Figure 20). This would avoid any unacceptable texture on a given surface of the implant. In the case where the implant has all three types of surfaces on its geometry, it would be necessary to machine the fracture surface first, followed by the periosteal and finally articular surface, in order to create the respective textures on these surfaces and maintain the implant's physiologic suitability.

3.7 Implementation And Results

The previously described algorithm for calculating surface specific setup orientations was implemented in C++ and an OpenGL user interface, and was tested on an Intel Core2Duo, 2.8 GHz PC, running Windows 7. The software accepts colored 2D slice files from 3D PLY models as input, and returns several analytical results. The minimum number of orientations necessary to create customized implant surfaces was calculated. Figure 21 and Table 5 show the process planning implementation for a specific bone implant with comparison of orientations from RM (*) and RM (◄). The orientations from RM are better aimed at individual surfaces than are the RM orientations. A new metric, *percent customization* is used to quantify the amount

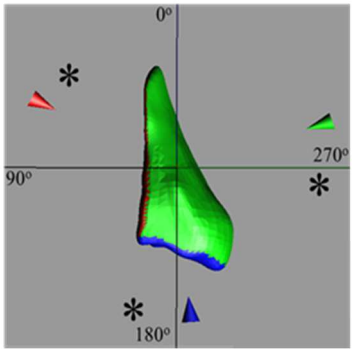


Figure 21: Setup orientations

of customized texture created on respective implant surfaces. This metric is driven by the percent tool path crossover and tool path redundancy, and measures the quantity of customized texture machined on implant surfaces. The normal deviation is used to optimize the roughness values of the machined texture. The customization algorithm always ensures an implant's suitability even if partially customized surfaces are created; for example, the tool path crossover from articular to periosteal surfaces maintains the implant suitability in spite of a reduced percent customization. Six models of varying complexities, types, and numbers of surfaces were used for calculating surface specific setup orientations (Figure 22). The analytical results (Table 6) show the percentage customization for each surface of each bone implant and the orientation computation time. A drop-tower was used to generate different samples of bone fragments.

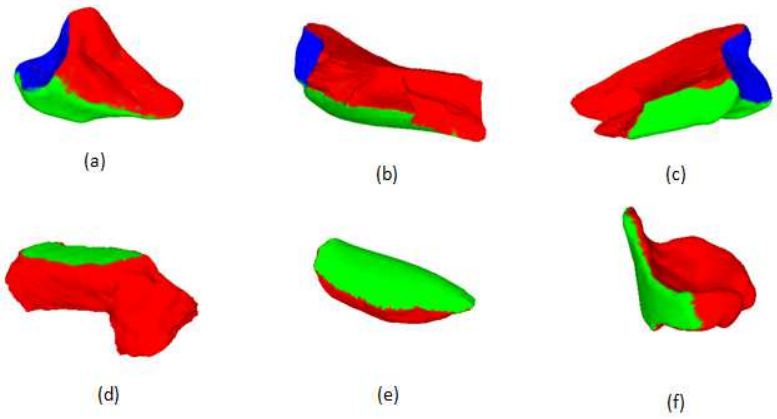


Figure 22: Different PLY models used for testing setup algorithm performance

These samples were subsequently CT scanned to generate CAD models. Individual surfaces were then manually colored and sent for machining using the RM process. Figure 23 shows three implants machined using RM orientations illustrating the rough texture on the fracture surface and respective smoother finishes on articular and periosteal surfaces. The percentage customization of each implant surface achieved

using orientations from the new methods is higher. On average, the new algorithm provided a 44 percent increase in customization of surfaces, with a minimum improvement of 8.5 percent to a maximum improvement of 68.7 percent (Figure 27). This shows that the new algorithm significantly increases percentage customization; for the bone implants, this successfully moves toward the goal of increasing fixation stability.

Table 5: Implementation results

Model	Facet count	RM (**No Customization**) Time (secs)	percentage customization of the surfaces Fracture	Articular	Periosteal
a	13214	4	8	98	85
b	80882	5	0	97	98
c	56182	4	13	0	76
d	11322	3	0	-	100
e	7574	2	82	-	100
f	10495	3	67	-	88
Model	Facet count	RM (**Customization**) Time (secs)	percentage customization of the surfaces (percentage increase) Fracture	Articular	Periosteal
a	13214	97	97(89)	100(2)	95(10)
b	80882	126	95(95)	100(3)	98(2)
c	56182	115	98(85)	100	97(21)
d	11322	85	46(46)	-	100(0)
e	7574	60	99(17)	-	100(0)
f	10495	82	96(29)	-	100(12)

Figure 23: Machined models (a, c, & f) with textured fractured surfaces

3.8 Machining Trials

An additional machining trial was conducted to compare the percentage customization calculated by the implemented software with the actual customization percentage imparted on the surfaces of the machined bone implant (Figure 24, 25, 26).

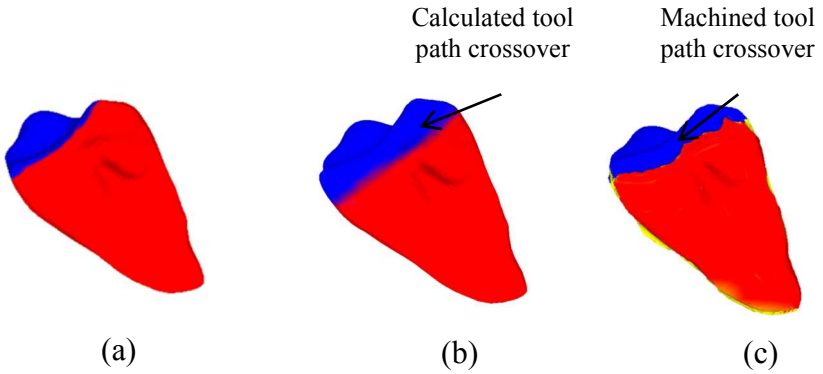


Figure 24: (a) Colored CAD model (b) Calculated fracture surface customization affected due to tool path crossover (blue) from articular surface specific orientation c) Detected fracture surface customization on machined implant affected due to Tool path Crossover (blue) from articular surface specific orientation

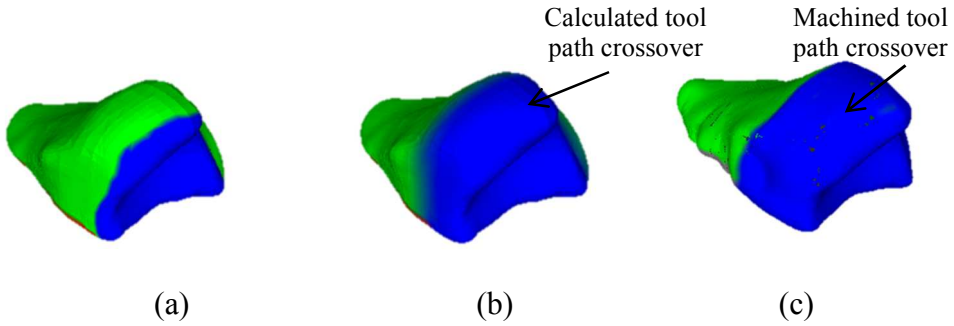


Figure 25: (a) Colored CAD model (b) Calculated periosteal surface customization affected due to Tool path Crossover (blue) from articular surface specific orientation c) Detected periosteal surface customization on machined implant affected due to Tool path Crossover (blue) from articular surface specific orientation

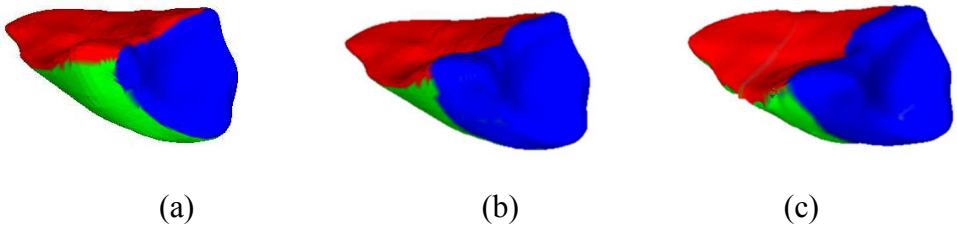


Figure 26: (a) Colored CAD model (b) Predicted customization of articular surface from articular surface specific orientation c) Detected customization of machined articular surface from articular surface specific orientation

Table 6: Customization percent comparison analysis vs machined model

Surface	% Customization	
	CAD model	Machined part
Fractured	92.89	91.54
Periosteal	89.12	91.04
Articular	100.0	100.0

3.9 Conclusion

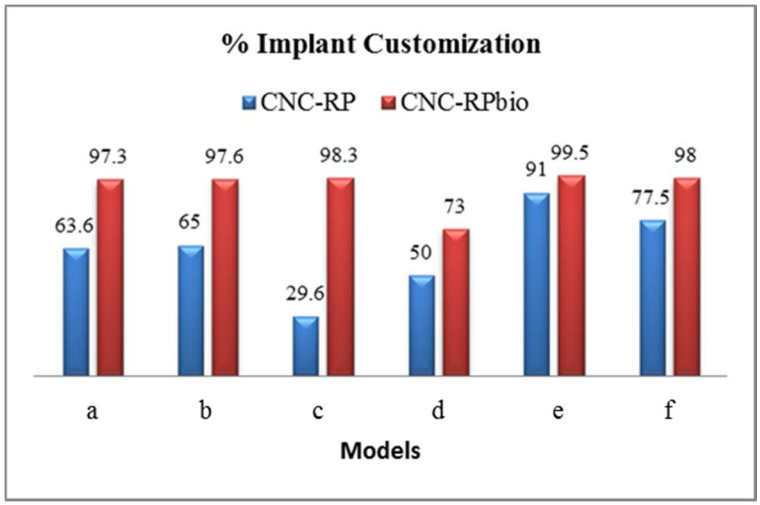


Figure 27: Setup CNC-RP vs. CNC-RP$_{bio}$

This paper introduces a new method for determining setup orientations to create defect-specific multiple surface parts with multiple custom textures using an automated rapid CNC machining process. The example of customized bone implant creation is presented. The implementation showed that the percentage customization for the implant surface textures achieved using the new algorithm is higher than algorithms that create implants with identical finish on all surfaces. This work illustrates how surface specific characteristics can be provided through targeting of surfaces on MSPs and then applying parametric changes to machining tool paths. These new setup processes have the potential to produce parts with improved performance, and in the case of implants, to consequently improve patient outcomes. Finding better setup angle solutions does extend processing time and

planning efforts (Table 6). That being said, however, this technique is still considered a rapid and highly automated method, since total process planning time (excluding the machining time) is still on the order of 30 to 45 minutes. Moreover, the modest increase in computation time is justifiable considering the significant potential improvement to part outcome, and in this example, to patient care.

3.10 Future work

The methods with specific modifications could be deployed for finish machining of selective critical surfaces using feature free cast models which would allow for creation of low tolerance surfaces on a cast part. Additionally, these methods can also be used for finish machining additively manufactured (AM) near net-shape components which require some level of post process machining. One proof-of-concept study coordinated with a collaborating lab was in the use of post-processing for Electron Beam Melting (EBM) components. EBM was used to create a near net-shape component in titanium, and then rapid machining was used to finish machine several (but not all) surfaces, in order to hold critical tolerances. Additionally the work presented in this paper could also prove beneficial towards manufacturing of legacy parts. CAD models derived from these parts using reverse engineering methods would generally be feature free. These feature free CAD models could algorithmically be processed further to detect individual surface clusters which could be assigned through color coding specific post processing or manufacturing methods. In the absence of the presented methods, one must rely on skilled machinists to create dedicated tool path plans, thereby reducing the “rapid” nature of the approach from the start.

3.11 References

[1] Frank, M.C., Wysk, R.A., and Joshi, S.B. (2006), "Determining Setup Orientations from the Visibility of Slice Geometry for Rapid CNC Machining," *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, Vol. 128, No. 1, pp. 228-238.

[2] Frank, M.C., Wysk, R.A., and Joshi, S.B. (2004), "Rapid Planning for CNC Machining – A New Approach to Rapid Prototyping," *Journal of Manufacturing Systems*, SME, Vol. 23, No. 3, pp. 242-255.

[3] Frank, M.C. (2007), "Implementing Rapid Prototyping Using CNC Machining (CNC-RP) Through a CAD/CAM Interface," *Proceedings of the Solid Freeform Fabrication Symposium*.

[4] Boonsuk, W. and Frank, M.C. (2009), "Automated fixture design for a rapid machining process," *Rapid Prototyping Journal*, Vol. 15, No. 2, pp. 111-125.

[5] Li, Y. and Frank, M.C., (2007), "Computing Non-Visibility of Convex Polygonal Facets on the Surface of a Polyhedral CAD Model," *Computer Aided Design*, Vol. 39, No. 9, pp. 732-744.

[6] LI, Y. and Frank, M.C. (2006), "Machinability Analysis for 3-axis Flat End Milling," *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, Vol. 128, No. 2, pp. 454-464.

[7] Thomas, T.P., Anderson, D.D., Willis, A.R., Lui, P., Frank, M.C., Marsh, J.L., Brown, T.D. (2011), "A computational/experimental platform for investigating three-dimensional puzzle solving of comminuted articular fractures," *Computer Methods in Biomechanics and Biomedical Engineering*, Vol. 14, No. 3, pp. 263–270.

[8] Willis, A., Anderson, D., Brown, T.T, Marsh, J.L. (2007), "3D Reconstruction of highly fragmented bone fractures," *SPIE Medical Imaging, Image Processing*, Vol. 6512, pp. 65121P.

3.11 Acknowledgements

This work was supported by grants from the NIH/NIAMS (P50AR055533 and R21AR054015), the Roy J. Carver Charitable Trust at the University of Iowa, State of Iowa Economic Development appropriations to the Board of Regents under the Grow Iowa Values Fund, and the Musculoskeletal Transplant Foundation. The authors would

also like to recognize the technical support of Mr. Gary Ohrt, and clinical inputs by Drs. Yuki Tochigi and J. Lawrence Marsh at the University of Iowa.

About the authors

Ashish Joshi received his PhD in Industrial Engineering at Iowa State University in 2014. He received his BS (2006) in Mechanical Engineering from Pune University (India), and his MS (2010) in Industrial Engineering from Iowa State University. His research interests are in the area of Advanced Process Planning for Rapid Manufacturing, Design for Manufacturing, CAM.

Matthew C. Frank, PhD, is an Associate Professor in the Department of Industrial and Manufacturing Systems Engineering at Iowa State University. Dr. Frank earned his BS and MS in Mechanical Engineering and a PhD in Industrial Engineering from The Pennsylvania State University. His research and teaching interests involve the general area of manufacturing, with a specific focus on rapid manufacturing and prototyping. He has published numerous papers and book chapters on RP and other manufacturing topics, has received the ISU E-Week Outstanding Professor Award for 2005, the 2006 SME Outstanding Young Manufacturing Engineer Award, and currently holds one US Patent.

Donald D. Anderson, PhD, is an Associate Professor at the University of Iowa in the Department of Orthopaedics and Rehabilitation, with a secondary appointment in Biomedical Engineering. Dr. Anderson holds a BSE in Biomedical Engineering, as well as an MS and a PhD in Mechanical Engineering (Biomechanics emphasis). During his twenty years plus of post-doctoral professional experience, Dr. Anderson's primary research focus has been in articular joint biomechanics, specifically investigating the

relationship between joint injury and subsequent development of post-traumatic osteoarthritis. Dr. Anderson is currently President of the American Society of Biomechanics, and also a member of the American Society of Mechanical Engineers, the Orthopaedic Research Society, the Orthopaedic Trauma Association, the Osteoarthritis Research Society International, and the International Society of Biomechanics.

Thaddeus Thomas, PhD, received his PhD in Biomedical Engineering from The University of Iowa in 2010. His research focused on developing computational methods for pre-operative orthopaedic surgical planning. He is now a civilian scientist working in the Army Research Laboratory, Weapons and Materials Research Directorate.

James Rudert, PhD, is a project engineer at the University of Iowa in the Department of Orthopaedics and Rehabilitation. Dr. Rudert holds BS and MS degrees in Mechanical Engineering, as well as a PhD in Biomechanical Engineering. His research focus is on design and fabrication of test equipment and experiments to determine physical properties of various orthopaedic devices and tissues.

Thomas D. Brown, **PhD,** is the Richard and Janice Johnston Chair of Orthopaedic Biomechanics at the University of Iowa, where he holds faculty appointments in the Departments of Orthopaedics and Rehabilitation (primary), Biomedical Engineering (secondary), and Mechanical Engineering (courtesy). His PhD (1976) is from Carnegie-Mellon University, in Mechanical Engineering - Bioengineering. Dr. Brown's research interests are in orthopaedic biomechanics and in clinical biomechanics of the musculoskeletal system. He is a past president of the American Society of

Biomechanics and of the Orthopaedic Research Society, and currently serves as Deputy Editor for Research for the Journal of Bone and Joint Surgery.

CHAPTER 4. AUTOMATED SETUP PLANNING FOR DISCRETE 3-AXIS MACHINING OF FEATURE FREE POLYGONAL MODELS

Ashish M. Joshi, MS, and Matthew C. Frank, PhD

Department of Industrial and Manufacturing Systems Engineering,

Iowa State University, Ames, IA 50010, USA

This paper presents a new process planning method for determining setups for discrete 3-axis machining of prismatic and freeform components. This paper specifically presents algorithms to determine a set of orientations for machining any given component using a generic 3-axis milling machine. The automatically determined machining orientations facilitate increased visibility and accessibility to part surface areas while optimizing on machining depths. The method in this paper uses constrained optimization with a Greedy approach and population based Genetic Algorithms (GA) to determine good setup solutions. To illustrate this utility, the setups solutions from the designed algorithms were used and the machining process was verified using Mastercam software. The verification showed that the determined setups and tool configurations (diameter and length) enable part surface creation with minimal non-accessible regions while optimizing on machining depths and visibility. To speed up the analysis, the algorithms were ported onto parallel processing hardware using an NVIDIA GPU C-2075, reducing processing time to a few minutes. The setup planning algorithms in this paper provide a means to create components to specifications while using cost effective, easily available and programmable 3-axis machining centers.

Keywords – Rapid Machining, visibility, machinability, parallel processing, GPU

4.1 Introduction

CNC-milling is a process of incrementally cutting material from a work-piece until a pre-determined geometry is created. This involves using a simultaneously advancing and rotating tool that performs the cutting operation (Figure 1). Traditional machining requires extensive planning that requires a skilled technician (machinist) to analyze part attributes like geometry, material composition (single/multiple), dimensions, tolerance, work piece, clamping, and available tool configurations. This also requires machinist to choose suitable machines and related parameters such as machining feed, speed. The time spent developing an optimal process plan for machining a component can be more readily justified in the case of mass production; however, custom designs or small lot sizes are significant motivation for automated process planning systems. Typically, CNC machines are available in different configurations viz. 3/4/5 axis (Figure 2). A standard 3-axis milling machine uses the Cartesian coordinate system having orthogonal X, Y and Z axis for machining a part. A 4-axis milling machine has an additional 4th rotary axis about X axis, while a 5-axis machine and yet another additional 5th rotary axis about the Y-axis. Rotary axis additions allow for additional visibility and accessibility to part surfaces. The part's visibility from a 3-axis machine is along the Z-axis in the X-Y plane, creating a 2½-D view space, while a 4th axis mill provides visibility about a cylindrical space and a 5th axis mill results in hemispherical visibility. This

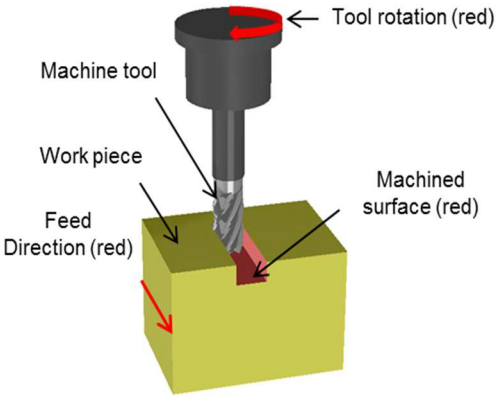


Figure 1: Milling process

means that reduced-axis configurations require the determination of multiple part setups and fixturing, which becomes a complex planning task depending on the part design. Hence, there is a tradeoff, where reduced-axis machines provide lower cost and ease of programming while more complex machines provide increased visibility.

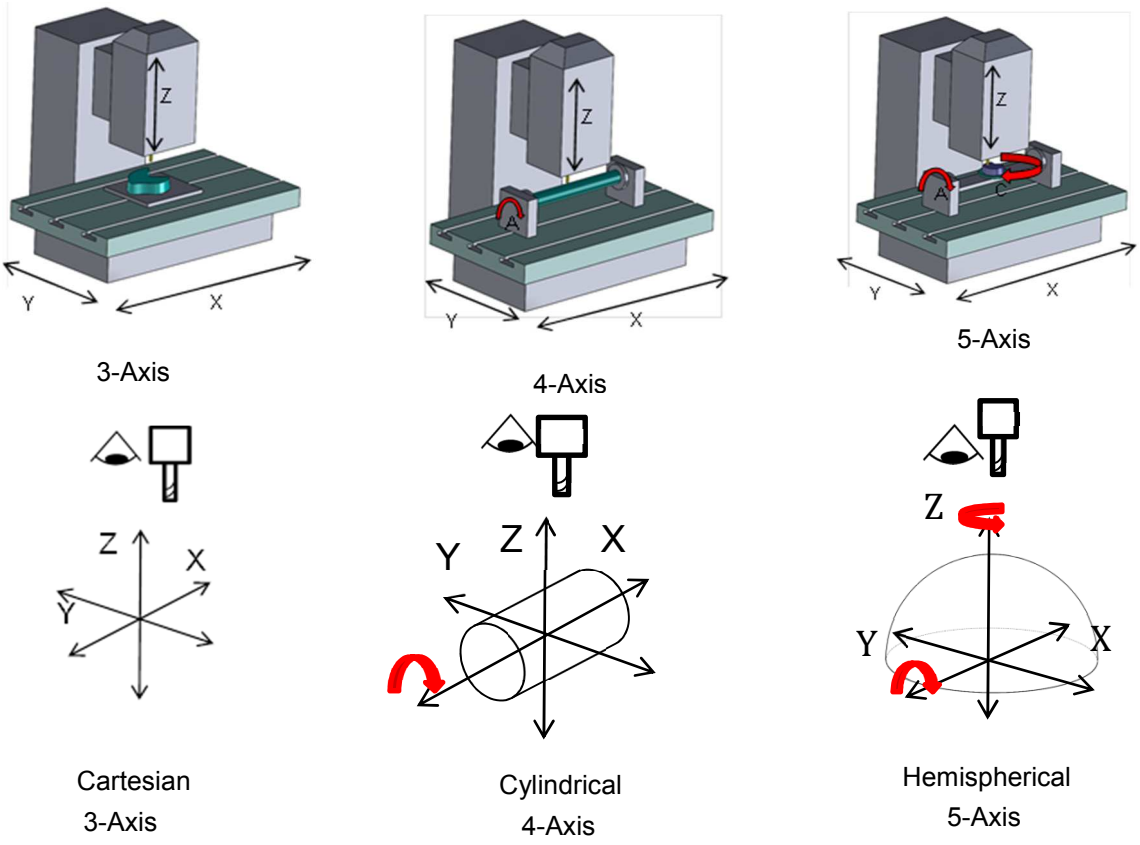


Figure 2: Milling machine configurations process

4.2 CAPP For 3-Axis Machining

The 3-axis milling configuration is a cost effective and usually preferable option; however, the lack of automated process planning systems leaves the machinist to manually determine part setups. Hence, this paper introduces algorithms that automatically determine a feasible set of orientations that could be used to machine a part. These algorithmic developments factor in **part visibility**, **tool accessibility** and

machining depths for analyzing different candidate orientations and tool selection that could affect geometric accuracy of the machined parts as well as efficiency of machining process (Figure 3).

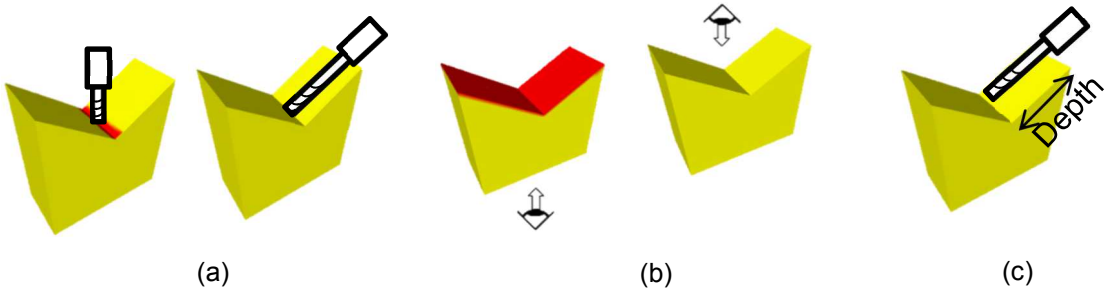


Figure 3: (a) Non-accessibility (b) Non-visibility(c) Machining depths

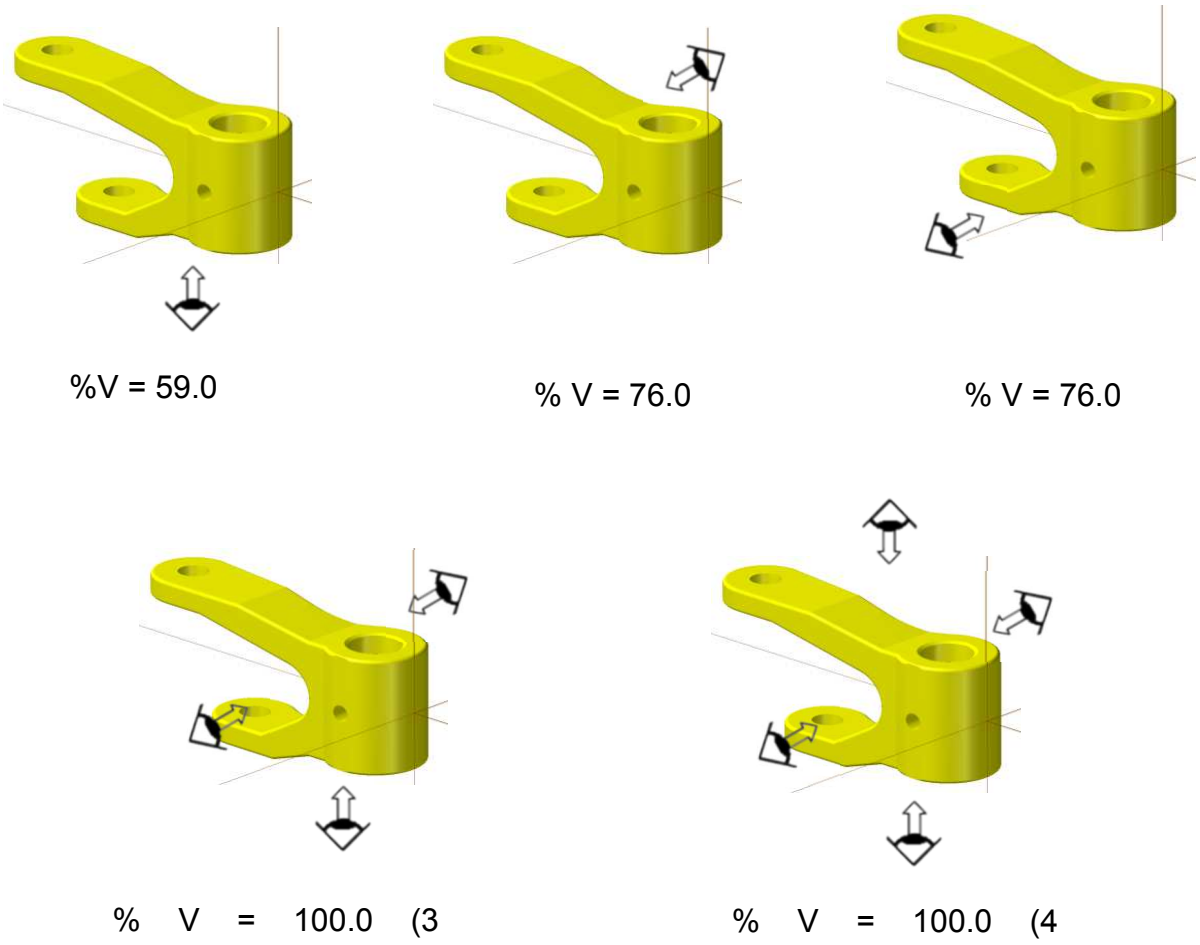


Figure 4: Line of sight Visibility (V)

4.2.1 Setup orientations

In order to machine a part, a set of setups is chosen where the part is machined from each of several orientations. Some of the many factors that determine a given setup's quality are the part's percent Non-Visibility (%NV) (line of sight), Non-Accessibility/Machinability (%NM) given a tool diameter, and machining depths or Non-Reachability (%NR) required for that setup.

4.2.2 Part non-visibility

The part non-visibility is based on line-of-sight visibility to the surface of the part from a given setup (Figure 4), where higher visibility per setup is better. The number of setups required for part machining can directly affect the cost of the part since more setups generally result in more time.

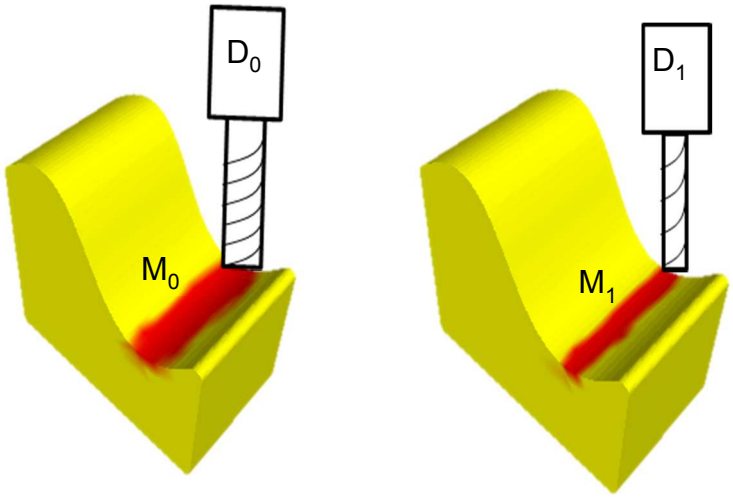


Figure 5: Tool diameter (D), $D_0 > D_1$, Machinability (M), $M_0 < M_1$, (red) Non-machinable area

4.2.3 Part non-machinability

The concept of machinability may have two connotations: 1) a material property that can describe how easy it is to physically cut a material, or 2) a geometric property about how easy it is to reach, access, and effectively create the part geometry. In this chapter the author solely considers the second description. This constraint or the inability of the tool to machine a complete part is largely dictated by its diameter. Figure 5 show that the given surface has higher percent machinability using a tool with smaller diameter as compared to a tool with larger diameter from the same setup.

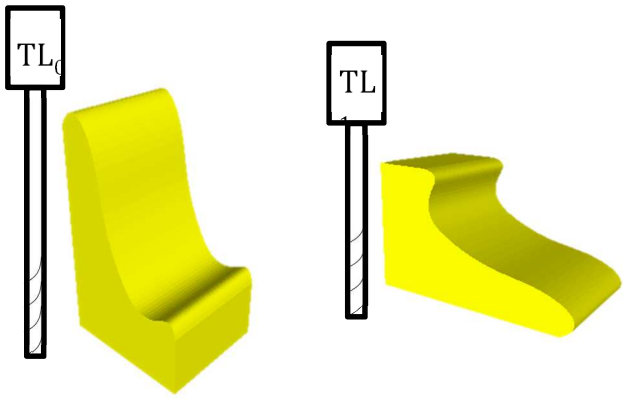


Figure 6: Tool Length (TL), $TL_0 > TL_1$

4.2.4 Part non-reachability/tool length

The concept of part reachability is based on the percentage of the part surface that can be contacted from a setup with a given tool length (Figure 6). In other words, this could be also considered as a parameter that dictates the tool length required to reach the part surface from a given setup. Since longer tools can be more problematic and force slower cutting operations, it is always better to use setups that provide shallower machining depths.

Hence considering together the above three concepts, a good setup is one which can provide maximum visibility, machinability and reachability (minimum non-visibility, non-machinability and non-reachability) for better surface quality and machining efficiency.

This paper focuses on determining setups that provide the optimum visibility, machinability and reachability enabling complete and cost effective machining of part using 3-axis milling.

4.3 Literature Review

Research in Computer Aided Process Planning (CAPP) for milling processes has been in existence since the late 1950's. With the advancements in milling processes and computing technologies, CAPP research was intended to reduce the manual planning efforts required from the skilled technicians and machinists. One of the most important process planning tasks for milling a part is to determine the setup orientations. The choice of orientations affects the machining time, tool path types, maximum feedrates possible, and geometric accuracy. Significant research in visibility maps and spherical algorithms were developed that could be used for setup planning for part machining [1][2]. In other work, a set of spherical polygons representing the visible curved surfaces were used to find a great circle (representing 4^{th} axis) and a band (representing 4^{th} and 5^{th} axes) containing a great circle that intersected the polygons maximally [3]. In [4], algorithms were developed to determine visibility based machining orientations of a 2-D slice file derived from 3-D polygonal models about a specific rotary axis. In [5], a feature based approach was used, where machining features were grouped into setups based on tool access direction and within each setup features were sequenced through geometric reasoning. In [6], another feature based approach was used where the number of setups were minimized by grouping them into classes of Single Approach Tool Directions (SATD) and Multiple Tool Approach Directions (MATD). In [7], feature based components that could be fixtured easily were considered. In this work, a robust

graph theoretic model of planning was presented along with a hierarchical prioritization of objectives in planning [7]. In [8], a tool selection strategy for 3-axis rough machining was chosen while in [9] algorithms were created for determining tool accessibility to part surfaces specific setup orientations. Setup planning, sequencing and other machining-related CAPP problems can also be effectively modeled as large-scale, combinatorial optimization problems; multi-objective optimization [10], Simulated Annealing [11], Ant Colony [12], Genetic algorithm [15][16][17][18][19][20][21], Hybrid Cuckoo search-genetic algorithm [13], Particle Swarm Optimization [14], have been applied successfully in CAPP optimization problems.

4.4 Manufacturing Using Rapid CNC Machining

The challenge of determining setups for discrete 3-axis rapid machining process is derived from previous work in rapid machining called CNC-RP (Figure 7). CNC-RP uses a standard 3-axis CNC milling machine with a 4th axis indexer for multiple setup orientations. This machining process includes completely automated fixture planning, tooling, and setup planning, including the generation of NC code for creating a part directly from a feature free CAD models (Frank et al., 2004, 2006, 2007). The use of a rotation axis eliminates the need for re-clamping of the part, a common task in conventional

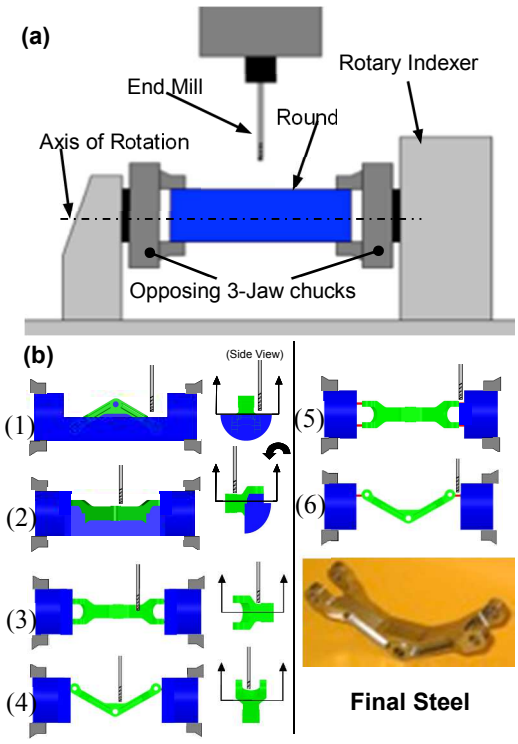


Figure 7 - (a) RM setup; (b) steps b.1-b.4 expose component geometry while b.5-b.6 exposes sacrificial supports

fixturing methods (Li and Frank, 2006, 2007). For each orientation, all of the visible surfaces are machined with reasonable tool configurations, while a set of sacrificial supports keeps the part connected to the uncut end of the stock material (Boonsuk et al, 2009). The algorithms presented in this chapter focus on determining setup orientations for discrete 3-axis configurations where we assume a single rotation axis is not sufficient for machining the entire part.

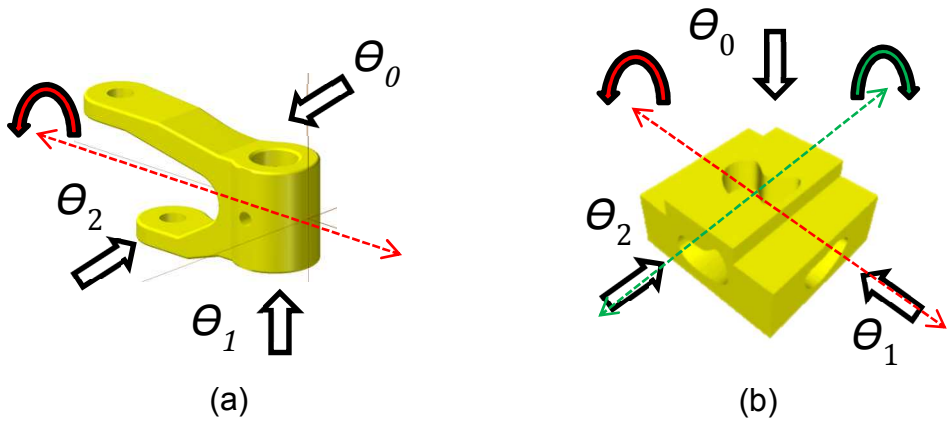


Figure 8: (a) Single axis setups (b) Multi-axes setups

4.5 Problem Statement

The objective of this paper is to develop CAPP methods for determining setup orientations for machining parts using discrete 3-axis machining configurations. The CAD input used is a slice model created from a polygonal model. The output is a set of orientations that should ensure complete machining of the entire part geometry.

4.5.1 Multi-axes setups

Oftentimes a part model is designed such that its geometry is not visible completely about any particular axis. In this case it becomes necessary to determine setups about multiple axes to completely machine the part. The use of a rotary indexer in the aforementioned CNC-RP process simplified the approach, where setups could be determined easily about 0°- 360° basis (Figure 7). In the case of more complex parts,

setups have to be determined using combinations of multiple axes for machining the part (figure 5). The methods presented can easily be extended to any number and types of combinations of axes. However, efforts to “design for manufacturing” , coupled with our conventional use of 3 planes in CAD modeling typically drive designs that have features accessible about three primary axes. Hence this chapter focuses on developing methods for determining setups about three primary axes X, Y, Z; where for any given model, the setup orientations would be 0°- 360° about any given axis.

4.5.2 Facet based analysis

Setup algorithms in this chapter use slice models created from polygonal models to provide setup solutions. Prior to determining the setups it is necessary to determine the part visibility. (Frank et al., 2004) presented a novel method to determine part visibility about a given axis using slice models. However due to uniaxial nature of slice models, part visibility cannot be extended onto other axes. Thus, in order to determine the part visibility about multiple axes a new facet based visibility method is presented that enables determination of setups about multiple axes. This method determines the visibility of each facet on the polygonal model, which enables multi-axis setup decisions. The facet based visibility technique can easily be extended to determine facet based machinability and reachability.

4.5.3 Meta-heuristics

The challenge of determining setups for a machining process can be described as a *Set Cover problem*. Set Cover problems fall in the NP-Hard category; which have no known algorithms that provide polynomial time solutions. In this research, we propose a Greedy Approach, combined with a 2-Level Genetic Algorithm (GA) in order to get a

feasible set of machining orientations. A good set of setups could be defined as one that; when machining completes on the final setup orientation, accurate part geometry has been efficiently created (complete geometry is revealed and the processing time is generally minimized).

4.5.4 Constrained optimization

As explained above, we argue that a good setup must provide minimum percentages for the parameters of Non-Visibility (%NV), Non-Machinability (%NM), and Non-Reachability (%NR). As could be expected in a multi-criteria problem, different setup solutions will provide different percentages for each parameter and no one setup solution provides a minimum percent for any of the parameters. This makes it necessary to use setups having a generic "goodness" defined by some combination of parameters. Thus the challenge in this chapter is modeled as a *constrained optimization problem* where the goodness measure of a set of orientations is evaluated through the designed objective function. We employ a nested or Multi-level optimization where %NV is optimized on Level-1 (L-1) and %NM and %NR is optimized on Level-2 (L-2) for a set providing minimum %NV. This multi-level approach is based on the assumption that setups providing minimum %NV are generally capable and worth minimizing %NM and %NR on. In other words, we propose that there is a priority order of importance among the three parameters. The objective function evaluated from L-2 and L-1 is combined together to get a single metric which defines the quality of a set of setups. Once the GA process is terminated according to a set criterion, the setup with the best metric is chosen as the solution.

4.5.5 Parallel computing using GPU

For determining setups, the use of 3-axes versus one increases the size of the decision space 3-fold and requires extensive analysis and significantly increases the time to get a solution. Thus, in order to increase the analysis speed, parallel processing hardware on an NVIDIA Graphical Processing Unit (GPU) was employed.

4.5.6 Chapter layout

The remaining sections of this paper are organized as follows. Section 4.6 presents a background on CAD input used for analyzing the part surface models in this work. Section 4.7 presents background on algorithms that are used to analyze the parameter values, while section 4.8 presents details on Multi-level optimization and explains the objective function that is used to evaluate goodness. Section 4.9 explains the use of a multi-level GA while section 4.10 explains the use of GPU programming. 4.11 shows implementations where setup orientations are determined for various parts and corresponding results are tabulated, and finally Section 4.12 presents conclusions and future work.

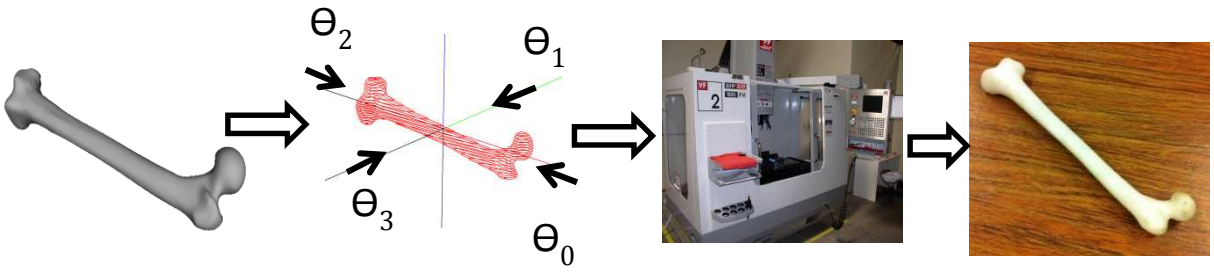


Figure 9: Subtractive Process

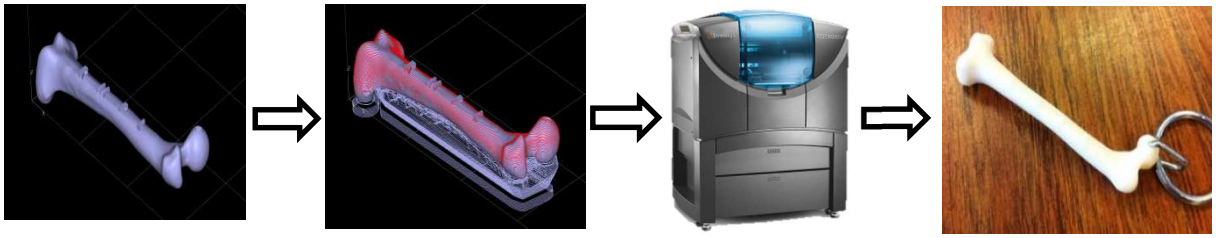


Figure 10: Additive Process

4.6 CAD Input

The setup planning methods presented in this paper uses 2D slice models generated from polygonal STL files. STL is the de-facto standard file type used by almost all modern 3D printers. An STL file consists of a triangulated surface representing a 3D CAD model, while the slice model derived from an STL model is a stack of 2D polygons (Figure 6). Slice models are used as input to rapid systems, where the part can be built slice-wise in an additive process, (Figure 4) or work piece material is removed slice-wise in a subtractive process (Figure 5). The setup planning methods presented in this paper use the concept of visibility, machinability, and depth analysis using slice models and previously developed algorithms [9, 21].

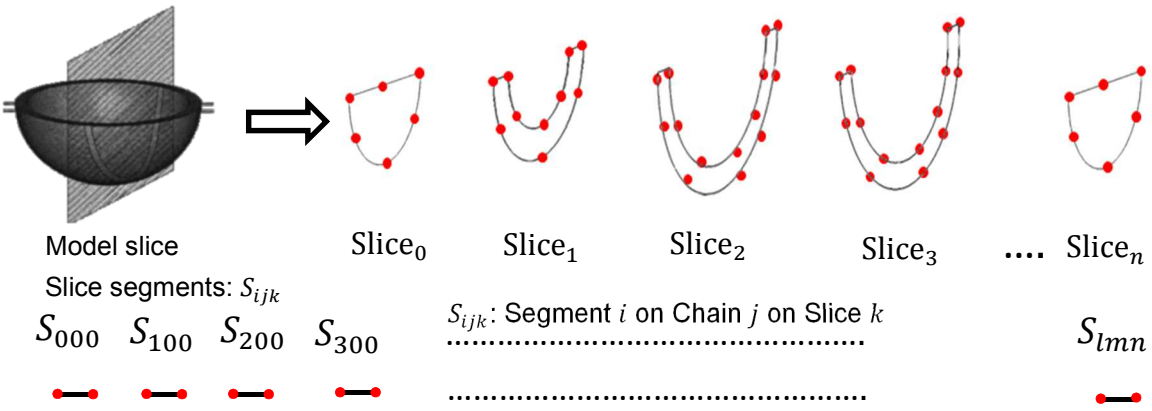
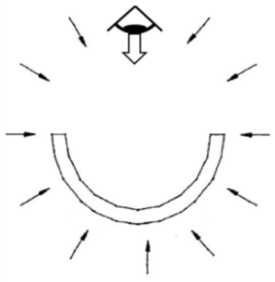


Figure 11: Slice Model

4.7 Non-Visibility (NV), Non-Machinability (NM), Non-Reachability (NR) Of A STL Model

4.7.1 Non-Visibility of a slice model

Algorithms to use slice models for CAPP of rapid machining were first developed by Frank et al [21] (Figure 6). Specifically, algorithms were developed for determining visibility-based setup orientations about a given rotary axis. In their work, line-of-sight based visibility for each segment was computed about a basis of 0°-360° and finally a set of machining orientations from which all the segments are visible was determined. The total percent NV of the STL part model about a given axis would thereby be approximated using the visible percentage of perimeters on the slices. Here, the visibility for a given segment is categorized into 2 types, a) local visibility i.e. the segment visibility with respect to the chain it belongs to on the same slice (Figure 8.a), and b) global visibility of the segment with respect to all the chains excluding the chain it belongs to (Figure 8.b). Hence using the visibility information of every segment on the slice model, and using a meta-heuristic approach, it is possible to determine the minimum number of orientations required for machining a part about a given rotary axis.

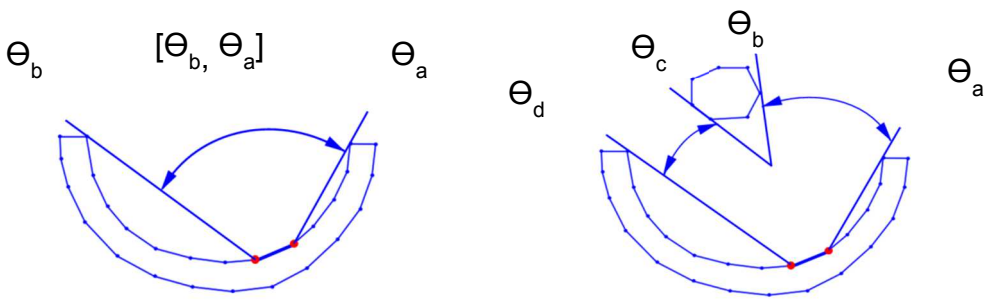


$$Visbility\ of\ segment: VS_{ijk}$$

$$Part\ Visbility\ V_p = \sum_{i=0}^{p}\sum_{j=0}^{q}\sum_{k=0}^{r}(VS_{ijk})$$

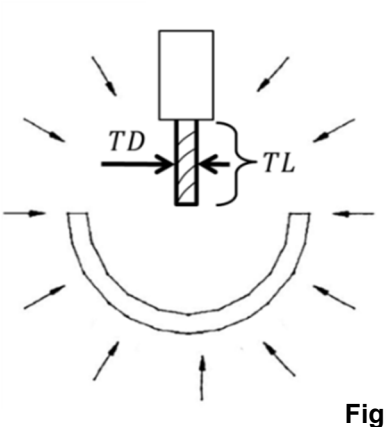
$$Non-\ Visbility\ NV_p = 1 - \sum_{i=0}^{p}\sum_{j=0}^{q}\sum_{k=0}^{r}(VS_{ijk})$$

Figure 12: Slice/Polygon Visibility



(a) Segment visibility [Θ_a, Θ_b] (b) Segment visibility [Θ_a, Θ_b], [Θ_a, Θ_b]

Figure 13: Segment visibility a) Local b) Global



$$Tool\ Diameter: TD$$

$$Part\ Machinability\ M_p = \sum_{i=0}^{p}\sum_{j=0}^{q}\sum_{k=0}^{r}(MS_{ijk})$$

$$Non-\ Machinability\ NM_p = 1 - \sum_{i=0}^{p}\sum_{j=0}^{q}\sum_{k=0}^{r}(VM_{ijk})$$

Figure 14: Non-Machinability

4.7.2 Non-Machinability of slice model

As explained previously, *machinability* is the ability of a tool, given its diameter, to contact all the regions of the part. The algorithm for determining part machinability was developed by Li et al [9]. In that approach, a slice model created from a polygonal model is used and machinability is determined for each segment present on the slice model, with tool diameter as input (Figure 12). The accessibility to each segment with a tool is analyzed considering other segments in the

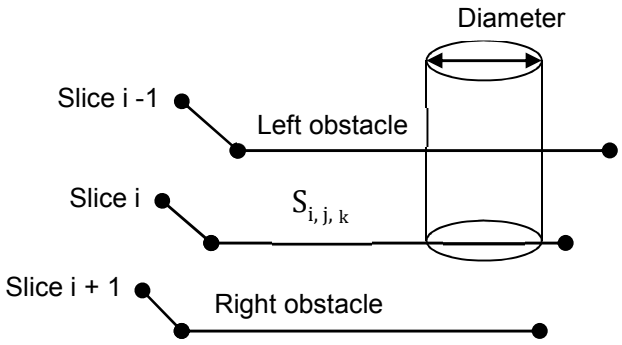


Figure 15: Non-Machinability of $S_{i,j,k}$

vicinity of the size of tool diameter (Figure 13). These segments in the vicinity potentially behave as obstacles and block the access to the segment of interest due to their location and orientation

4.7.3 Non-Reachability of slice model

Non-Reachability of a part model is simply the ability of the tool to reach a specified machining depth. This ability depends on the length of the tool and determines the percent of surface area a tool could reach for a given setup. It is always best to have a minimum NR or maximum reachability for a tool from a given setup. One problem with using longer tools to reach is that long skinny tools can deflect and or chatter easily. Although one could choose a tool with larger diameter in order to reach farther, this would have an obvious detrimental effect on Machinability. Hence setups must be chosen that provide reachability with shorter tools. The reachability for a part model can be determined by using an input slice model where the maximum depth can be determined by calculating 2D distances for all the segments visible/machinable from a specific setup about the axis of rotation.

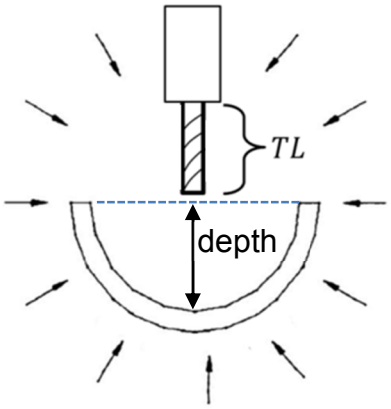


Figure 16: Reachability

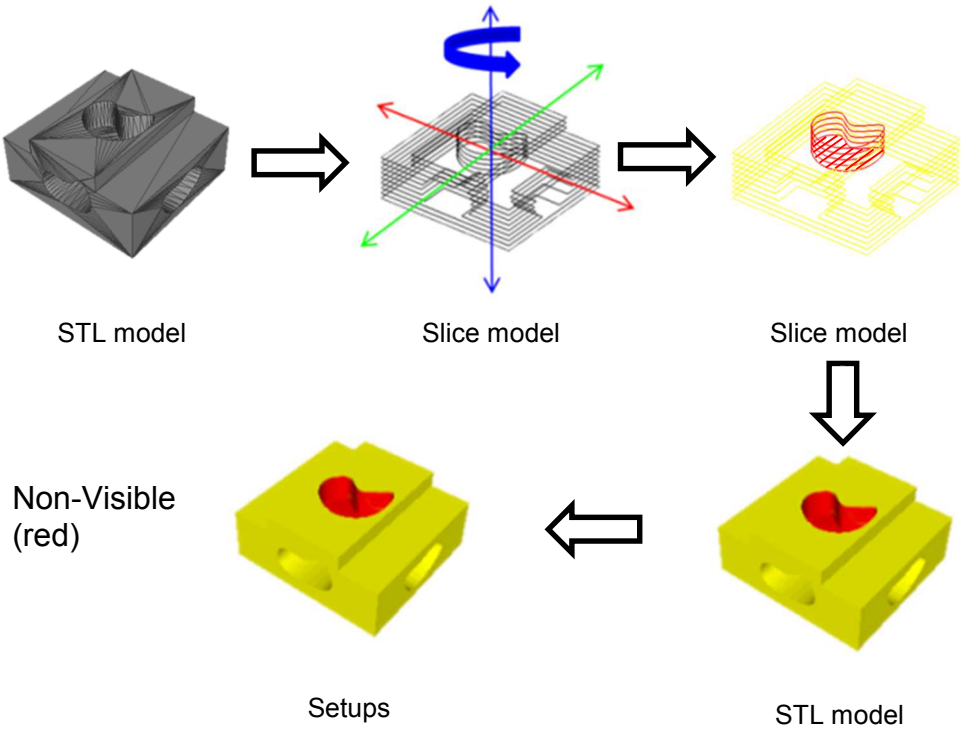


Figure 17: Non-Visibility about single axis (blue)

4.7.4 Mapping results from slice model to STL model

When it comes to multi-axes machining of parts, It becomes not only necessary to detect the NV, NM and NR regions about a particular axis but also to determine orientations about other axes that would allow machining of the same regions, perhaps more easily. Due to the requirement for analyzing the model about three primary axes, we present a method that allows mapping of analysis results back to the STL model from slices (Figure 17). In this manner, visibility of the entire STL model can be determined by accumulating the visibility of each facet originally determined through the analysis of its segments on the slice model (Figure 8). It is then possible to determine the visibility, machinability and reachability of each facet on an STL model about any given axis (Figure 10).

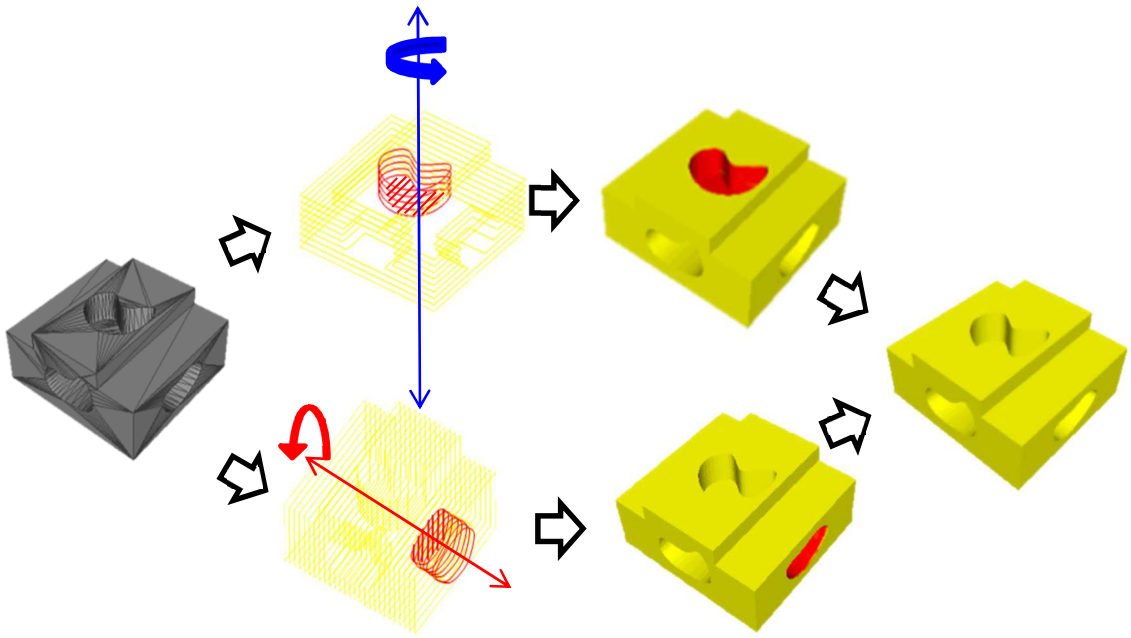
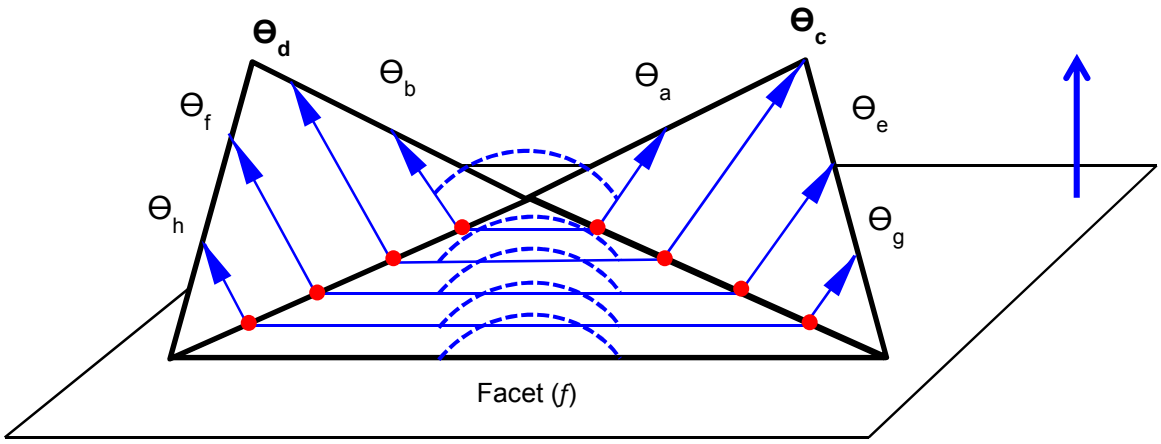


Figure 18: Combined Non-Visibility about multiple axes (blue, red)

4.7.5 Visibility analysis of a facet

In order to determine setups about multiple axes, it is necessary to determine the visibility of the STL model, which requires visibility data for each facet. Hence a new method is presented where the visibility of each facet is deduced by clubbing the visibility information of all the segments originating from their corresponding facets (Figure 9). In this manner, the visibility of every facet could be deduced about all 3 axes by analyzing the corresponding slice model and mapping back the visibility data from segments to their parent facets. Similar to the visibility analysis, the results from the machinability and reachability analysis performed on slice model segments can be mapped back to their corresponding facets also. In this manner the total % Non-Visibility, Non-Machinability, and Non-Reachability can be determined for a STL model about multiple axes with provided tool configurations.



Segment visibility $Sg_{i,j,k}$: $[\Theta_a, \Theta_b]$, $Sg_{i,j,k+1}$: $[\Theta_c, \Theta_d]$, $Sg_{i,j,k+2}$: $[\Theta_e, \Theta_f]$, $Sg_{i,j,k+3}$: $[\Theta_g, \Theta_h]$

Facet (*f*) visibility: $[\Theta_a, \Theta_b] \cap [\Theta_c, \Theta_d] \cap [\Theta_e, \Theta_f] \cap [\Theta_g, \Theta_h] = [\Theta_a, \Theta_b]$

Figure 19: Facet (*f*) visibility

4.8 Constrained Multi-level Optimization

This section presents the constrained multi-level optimization and the designed objective function used to evaluate the goodness measure of a set of orientations for 3-axis milling (Figure 20). A minimizing objective function is designed that combines the parameters of; 1) Non-Visibility (% NV), 2) Non-Machinability (%NM), (3) Tool Diameter (TD), and 4) Tool Length (TL). The objective function has been divided into three levels, where each level is nested within the previous one (Figure 20). In level 3, the objective function minimizes the total percent Non-Machinability (%NM) out of the total Non-Visibility from a given setup given a tool diameter and length. This allows choosing an appropriate diameter while minimizing %NM for a given setup. The evaluation metric from Level-3 is plugged into Level-2 where the total percent Non-Visibility (%NV) is determined in addition to the tool length required to reach. In this manner, the % NM, %NV and TD and TL are optimized per setup. The metrics evaluated per setup from

Levels 3 and 2 are plugged into the Level-1 objective function that houses the parameters of total Non-Visibility and Non-Machinability for the part and total number of setups and setup axes required to machine the part. In summary: Level-3 focuses on minimizing %NM per setup through the use of appropriate TD, Level-2 focuses on minimizing machining depths through the use of appropriate tool lengths, and Level-1 forms the top-most level where in addition to results from Level 2 and 3, the number of setups and axes required are considered. The metric from Level-1 provides a comprehensive *quality* of a given set of setups for a given part.

The machining time depends directly on machining speeds and feed, which is affected by tool stiffness, based on its diameter and length, and can also affect surface finish. Related research presented concepts on the effects of tool configurations on machining time [8]. It was shown that the tool diameter (TD) directly affected the Material Removal Rate (MRR), while the tool length (TL) inversely dictated the MRR, where they needed to limit the bending stresses and avoid tool breakage.

$$MRR\ \alpha\ {}^{TD}/_{TL}$$

In this work, we will attempt to add tool diameter and tool length as parameters used in our optimization. The overall objective now becomes; machine the part with accurate geometry such that the Non-Visibility, Non-Machinability, Tool Length are minimized and Tool Diameter is maximized. Since tools are made in a limited set of lengths, we use a discretization of the total machining depths to be considered. Figure 9 shows the

visibility from a specific orientation, where the total depth is discretized into smaller regions. In figure 10 appropriate tools are chosen for each discretized depth.

$$(NV_P)_\Theta = \sum_{d=0}^{dmax} \left((NV_p)_\Theta\right)_a$$

$dmax$ = d_1+ d_2+ d_3

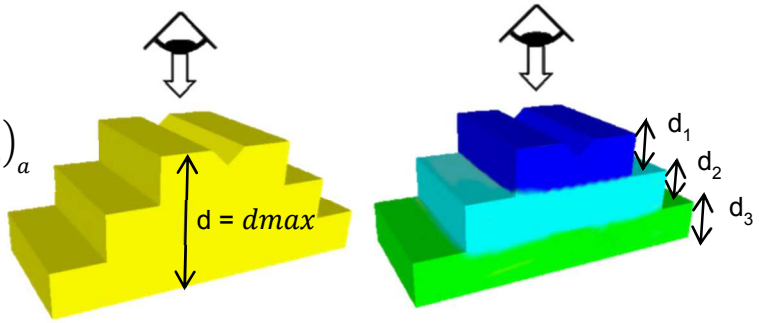


Figure 20: Minimize percent Non-Visibility per depth

$$(NM_P)_\Theta = \sum_{d=0}^{dmax} \left((NM_p)_\Theta\right)_{d,a}$$

$dmax$ = d_1+ d_2+ d_3

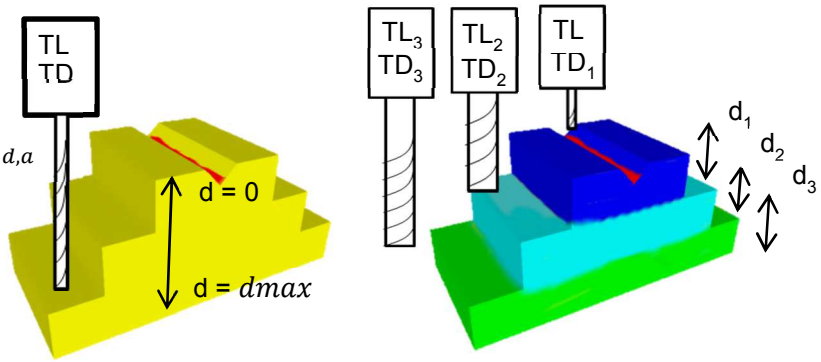


Figure 21: Minimize percent Non-Machinability per depth

Additionally, we generally prefer setups that require the part to be oriented about a minimum number of setup axes. Using multiple axes requires a machinist to plan for complicated fixturing solutions which could result in compromised part accuracy. Hence the objective function in Level-1 also evaluates the number of setup axes about which orientations are considered.

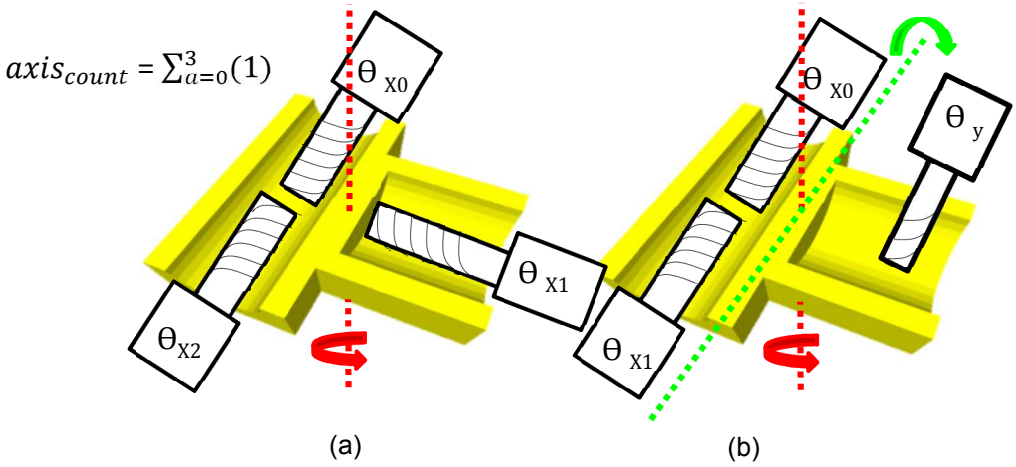


Figure 22: Setups about (a) single setup axis (red) (b) multiple setup axes (red,

The setup orientation count is used as a parameter in the objective function at Level-1 that focuses on minimizing the number of orientations while giving primary importance to parameters like Non-Visibility, Non-Machinability, Machining depths and tool configurations selection.

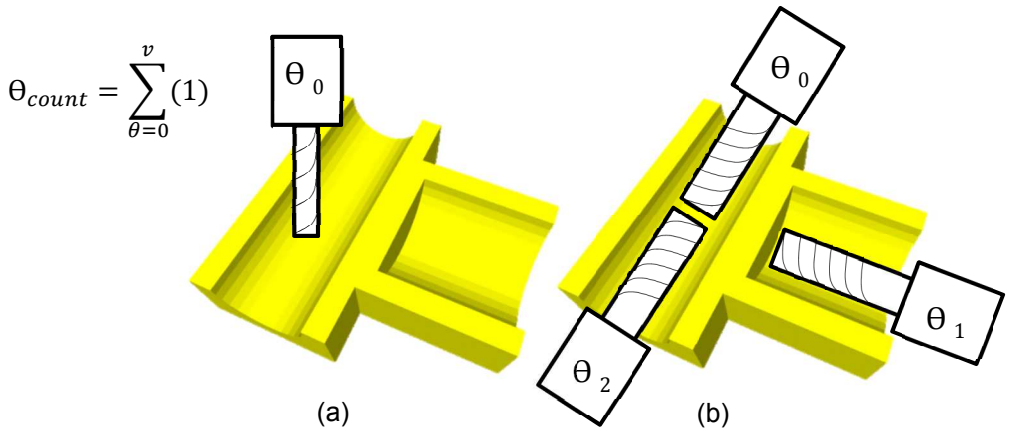


Figure 23: Orientation count (a) single (b) multiple

The aforementioned parameters, when combined, could be used as a part of a single objective function. This objective could evaluate different sets of machining orientations and allow the machinist to choose the optimum set that would make the part with high geometric accuracy. The following objective function thus provides a metric that

evaluates the optimality of a candidate orientation set. Other variable used in this optimization procedure are given in the appendix.

$$Min\, f(\Theta) = Min\, f(\Theta_0, \Theta_1, .., \Theta_w)$$

Objective/Fitness function

$$\sum_{\Theta=\Theta_0}^{\Theta_n} \sum_{d=0}^{(d_\Theta)max} Min\{\emptyset(NV, TL)_{\Theta,d}\} + \mu(\%NV)_P \longrightarrow \textbf{Non-Visibility (C-1)}$$

$$\text{Min}\{f(\Theta)\} = + \sum_{\Theta=\Theta_0}^{\Theta_n} \sum_{d=0}^{(d_\Theta)max} Min\{\varphi(NM, TD)_{\Theta,d}\} + \varepsilon(\%NM)_P \longrightarrow \textbf{Non-Machinability (C-2)}$$

$$+ \sigma(axis_{count}) + \psi(\Theta_{\text{count}}) \longrightarrow \textbf{(C - 3)}$$

$$\text{Min}\{f(\Theta)\} = (C1) + (C2) + (C3)$$

Level-1 $Min\{\emptyset(NV, TL)_{\Theta,d}\}$	**Evaluate Non-Visibility (C-1)** $\sum_{\Theta=\Theta_0}^{\Theta_n} \sum_{d=0}^{(d_\Theta)max} Min\{\emptyset(NV, TL)_{\Theta,d}\} + \mu(\%NV)_P$ $Min\{\emptyset(NV, TL)_{\Theta,d}\} = \left\{ \gamma \left(NV_{\Theta,d}\right)/_{NV_\Theta} + \delta(\text{TL})_d \right\}$
Level-2 $Min\{\varphi(NM, TD)_{\Theta,d}\}$	**Evaluate/Optimize Non-Machinability (C-2)** $\sum_{\Theta=\Theta_0}^{\Theta_n} \sum_{d=0}^{(d_\Theta)max} Min\{\varphi(NM, TD)_{\Theta,d}\} + \varepsilon(\%NM)_P$ $Min\{\varphi(NM, TD)_{\Theta,d,a}\} = \left\{ \alpha \left(NM/_{1-NV}\right)_{\Theta,d,a} + \beta\, TD_{a,d} \right\}$

Figure 24: Multi-Level optimization

Objective function variables

$\Theta = [\Theta_0, \Theta_1, .., \Theta_w]$ Set of setup orientations

(NV) Non-Visibility

(NM) Non-Machinability

$(TA)_P = \sum_{f=0}^{u} (TA)_f$ Total part area

$\%(NV)_P = \left(\left({}^{(NV)_P}\!/_{(NV)_{min}} \right) \times 100.0 \right)$ % Non-Visibility of part

$\%(NM)_P = \left(\left({}^{(NM)_P}\!/_{(NM)_{min}} \right) \times 100.0 \right)$ %Non-Machinability of part

$(NV)_P = \sum_{f=0}^{u} (NV)_f$ Non-Visible part area

$(NM)_P = \sum_{f=0}^{u} (NM)_f$ Non-Machinable part area

$(NV_P)_\Theta = \sum_{d=0}^{dmax} \left((NV_p)_\Theta \right)_d$ Non-Visible part area from a setup

$(NM_P)_\Theta = \sum_{d=0}^{dmax} \left((NM_p)_\Theta \right)_{d,a}$ Non-Machinable part area from a setup

$(TA)_f$ Total area of facet f

$(NV)_f = \sum_{i=0}^{p} \sum_{j=0}^{q} \sum_{k=0}^{r} (NV_{S_{ijk}})_f$ Total percent Non-Visible facet

$(NM)_f = \sum_{i=0}^{p}\sum_{j=0}^{q}\sum_{k=0}^{r}(NM_{S_{ijk}})_f$	Total percent Non-Machinable facet
$\left((NV_p)_\theta\right)_d$	%NV of a part p from a setup θ
$\left((NM_p)_\theta\right)_{d,a}$	%NM of a part p from a setup θ
$(NM_{S_{ijk}})$	Non-Machinability status of
$(NV_{S_{ijk}})$	Non-Visibility status of $S_{i,j,k}$
$(S_{i,j,k})_f$	$S_{i,j,k}$ belonging to Facet f
$S_{i,j,k}$	Segment i on chain j on slice k
$\theta_{count} = \sum_{\theta=0}^{v}(1)$	Number of setup orientations in a candidate set
$axis_{count} = \sum_{a=0}^{3}(1)$	Number of setup axes in a candidate set

Indices	Bounds
θ : Setup orientation	*u* : Facet count
f : Facet index	*l* : Segments
i: Segment	*m* : Chains
j: Chain	*n* : Slices
k: Slice	a_{max} : Max tool diameter
a: Tool Diameter	l_{max} : Max tool length
l: Tool Length	d_{max} : Max machining depth
d: Machining depths	
$\{\theta \in N \mid 0 \le \theta < 360\}$	$\{k \in N \mid 0 \le k < n\}$
$\{f \in N \mid 0 \le f < u\}$	$\{a \in R \mid 0.125 \le a < a_{max}\}$
$\{i \in N \mid 0 \le i < l\}$	$\{l \in N \mid 0 \le l < l_{max}\}$
$\{j \in N \mid 0 \le j < m\}$	$\{d \in N \mid 0 \le l < d_{max}\}$

4.9 2-Level Genetic Algorithm (GA)

Genetic Algorithms (GA) are population based meta-heuristics that allow exploration and exploitation of a design space until a global optimum is found or a termination criterion is satisfied. In this paperwork, GA is used due to the fact that determining setup orientations for machining process is an NP-HARD problem, where determining a global optimum solution may not be possible within polynomial time. Oftentimes solutions close to local optimums are provided by meta-heuristics that may or may not be sufficient. However in the realm of machining, good and feasible solutions are acceptable as long as the part is created according to specified design with an optimized machining process. A two level GA is used, where, on Level-1 (L-1) the GA explores and exploits the design space for minimizing percent Non-Visibility (%NV) and Tool Length (TL) amongst a population of set of orientations. On Level-2 (L-2) the GA explores and exploits another design space for minimizing percent Non-Machinability (%NM) and Tool Diameter (TD) amongst a population of set of Tool Diameters. As explained in the previous section Table 1 shows the objective function divided into multiple components. For a set of setups, component C-1 evaluates %NV and TL, while the component C-2 evaluates %NM and TD. Finally, the number of setup axes and setups is the component C-3. The components C-1, 2, and 3, all affect the geometric accuracy and cost of a part. However components C-1 and 2 are more related to the machining process while C-3 is related to part fixturing. Hence in this paper the author focuses on creating the part accurately by optimizing C-1 and C-2. Though C-3 is not optimized in this paperwork, it is considered as part of the objective function and contributes to its goodness measure.

Table 1: Genetic algorithm layout

<table>
<tr><td colspan="2">Objective/Fitness function
$\sum_{\theta=\theta_0}^{\theta_n}\sum_{d=0}^{(d_\theta)max} Min\{\emptyset(NV,TL)_{\theta,d}\} + \mu(\%NV)_P \longrightarrow$ Non-Visibility (C-1)
$\mathrm{Min}\{f(\theta)\} = + \sum_{\theta=\theta_0}^{\theta_n}\sum_{d=0}^{(d_\theta)max} Min\{\varphi(NM,TD)_{\theta,d}\} + \varepsilon(\%NM)_P \longrightarrow$ Non-Machinability (C-2)
$+ \sigma(axis_{count}) + \psi(\theta_{\mathrm{count}}) \longrightarrow$ (C - 3)
$\mathrm{Min}\{f(\theta)\} = (C1) + (C2) + (C3)$</td></tr>
<tr><td>Level-1 GA
$Min\{\emptyset(NV,TL)_{\theta,d}\}$</td><td>Evaluate Non-Visibility (C-1)
$\sum_{\theta=\theta_0}^{\theta_n}\sum_{d=0}^{(d_\theta)max} Min\{\emptyset(NV,TL)_{\theta,d}\} + \mu(\%NV)_P$
$Min\{\emptyset(NV,TL)_{\theta,d}\} = \left\{ \gamma \left(NV_{\theta,d}\right)/_{NV_\theta} + \delta(\mathrm{TL})_d \right\}$</td></tr>
<tr><td>Level-2 GA
$Min\{\varphi(NM,TD)_{\theta,d}\}$</td><td>Evaluate/Optimize Non-Machinability (C-2)
$\sum_{\theta=\theta_0}^{\theta_n}\sum_{d=0}^{(d_\theta)max} Min\{\varphi(NM,TD)_{\theta,d}\} + \varepsilon(\%NM)_P$
$Min\{\varphi(NM,TD)_{\theta,d,a}\} = \left\{\alpha\left({}^{NM}/_{1-NV}\right)_{\theta,d,a} + \beta\, TD_{a,d}\right\}$</td></tr>
</table>

4.9.1 Level-1 (L-1) Genetic Algorithm (GA)

On L-1 the GA is employed such that it minimizes the %NV and TL amongst a population of set of orientations for a given part. The set of orientations providing combined minimum of %NV and TL's is sent to Level-2 for which %NM and TD's are minimized for each setup. The details for the L-1 GA including parameters and corresponding operations are shown in figure 25.

$\{l \in N \mid l > 0\}$ Setups (0 - m) →

$$\begin{bmatrix} A_0 \\ A_1 \\ . \\ . \\ A_l \end{bmatrix} = \begin{bmatrix} (\theta_0, \theta_1, \ldots\ldots, \theta_m)_0 \\ (\theta_0, \theta_1, \ldots\ldots, \theta_m)_1 \\ . \\ (\theta_0, \theta_1, \ldots\ldots, \theta_m)_l \end{bmatrix}$$

Parameter

Phenotype: θ = 0° – 359°

Phenotype set size: 2 - 6

Genotype: θ_0 = 90.0 = 1011010

Encoding: Binary

Population size: 25

Generations: 20

Operator

Reproduction/Selection: Elitism

Crossover: 5 bit (Max 31° deviation)

Crossover rate: 0.75 (75%)

Mutation: 6th Bit flip

Mutation rate: 0.1 (10%)

Figure 25: Level-1: GA Optimizing C-1

4.9.1.1 Seed population

To run the GA, a seed population is used upon which operations are performed to determine the optimum solution. The seed used in the GA on L-1 is a population of sets of orientations designed by the author based on two major considerations, viz. number of setups affecting machining costs and placement of setups around the part affecting the ability to machine the part. These considerations are elaborated as follows: At least

two setups are necessary to machine a part; however, there could be an infinite combination and number of orientations used to machine the same part. In a 3-axis machining process, the total time is influenced by the actual machining time and stock re-fixturing every time the setup is changed. Hence, on L-1 the minimum number of setup in a set used in the seed are two, while maximum number of setups considered is six. We assume that two to six setups are usually feasible and reasonable for a machining process depending on part's design complexity and dimensions (Figure 26).

In addition to the number of setups the placement of these setups around the part is also affected by the part design. Generally, parts are designed such that the part is visible from orthogonal angles, allowing for easy accessibility and re-fixturing (Figure 27). However there are always exceptions where complicated designs or freeform shapes place part features oriented at non-orthogonal angles. Thus

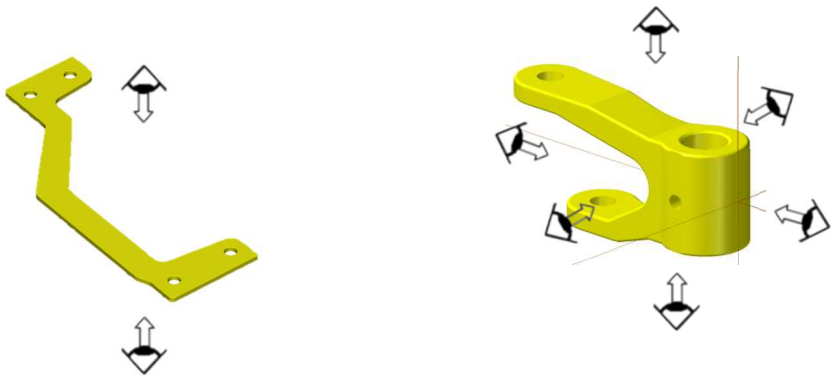


Figure 26: (a) two (b) four setups

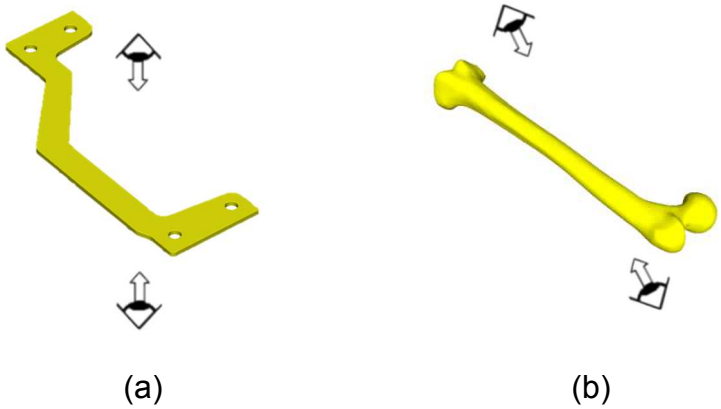


Figure 27: (a) Orthogonal (b) Non-orthogonal setups

the above considerations were taken into account while designing seed sets for L-1 GA. The seed population used is shown in the implementation section.

4.9.1.2 Encoding

$$(0,90,180,270) \quad = \quad (00000000, \quad 1011010, \quad 10110100,$$

Figure 28: Binary encoding

The encoding used in the L-1 GA is binary where every set of setup orientations or a chromosome in the seed population was converted to a string of bits, 0 or 1 (Figure 28). The input to the GA at L-1 is a population of feasible sets of setup orientations generated using Greedy algorithms. These feasible sets are improvised upon by the GA on 2 levels. The Level-1 (L-1) optimizes on the part visibility by choosing the best set of setups among the population. Once on L-1, a set of setups based on visibility and machining depths is chosen and the same set is optimized by considering machinability for multiple tool configurations on Level-2 (L-2). In this way, the goodness measure of a given set of orientations is determined using a nested objective function, the parameters for which are determined by 2-Level GA.

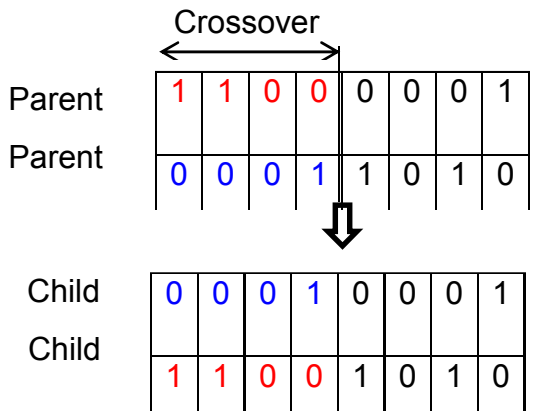


Figure 29: Crossover

4.9.1.3 Crossover

The crossover is a genetic operation in GA to vary the genetic information of the chromosome. The intent in this operation is to swap the information of parent chromosomes (Figure 29) with the hope to produce higher quality offspring. The crossover method used in L-1 GA in this paperwork is a 5-bit crossover where, for a given set of parents, the crossover site chosen is at the 5th bit for each binary encoded setup in the chromosome (Figure 30). The motivation for 5th bit as the crossover site was to allow maximum variation of 31° for any given setup in the chromosome. This provides enough variation in setup range and allowing for exploitation of the neighborhood in the entire design space. The crossover rate used in L-1 is 0.5.

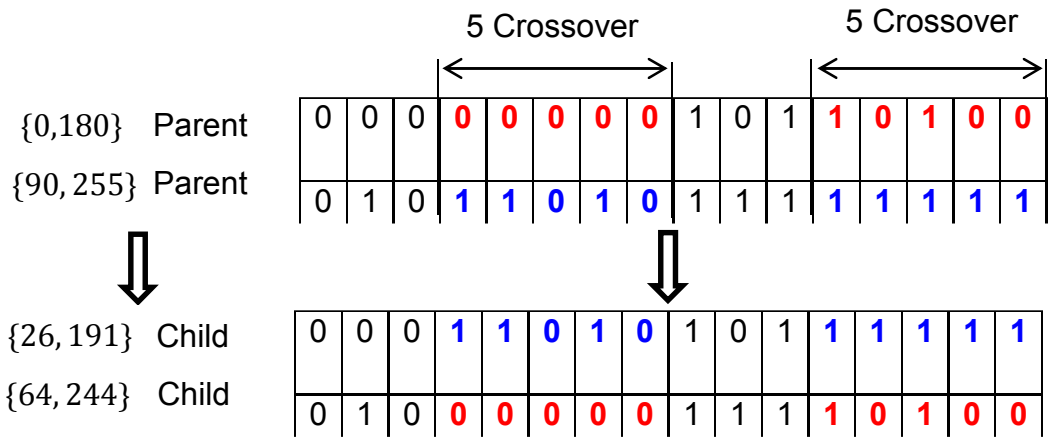


Figure 30: L-1 GA crossover

4.9.1.4 Mutation

Mutation is the genetic operation in GA which is similar to crossover changes of genetic information of the chromosome. Unlike crossover in mutation, selective bits in the binary encoded chromosomes could be flipped in order to change their value (Figure 31). This

allows sufficient exploration of the design space by changing the variable or chromosome information significantly. In this paperwork, the mutation operation is designed to flip the 6th bit of every setup encoded in a chromosome. This prevents the solution from getting stuck in local minima and allows exploration of the design space by changing setup values significantly. The mutation rate used in L-1 was 0.1.

$$A_l = \begin{bmatrix} \theta_0 \\ \theta_1 \\ . \\ \end{bmatrix}_l = \begin{bmatrix} (TD_0)_0 \dots\dots\dots\dots\dots\dots (TD_0)_n \\ (TD_1)_0 \dots\dots\dots\dots\dots\dots (TD_1)_n \\ \vdots \\ (TD_m)_0 \dots\dots\dots\dots\dots\dots (TD_m)_n \end{bmatrix}$$

Parameters

Phenotype: TD = 0.125 - 0.5 $\{m \in N \mid 0 \leq m < 360\}$

Genotype: 0.125 - 0.5 $\{n \in N \mid 0 \leq m < 10\}$

(TD): Inches

Encoding: Value based

Population size: 6

Generations: 15

Operators

Reproduction/Selection: Elitism

Crossover: Random 2-site

Crossover rate: 0.75 (75%)

Mutation: Value flip

Mutation rate: 0.1 (10%)

Figure 31: Level-2: GA Optimizing C-2

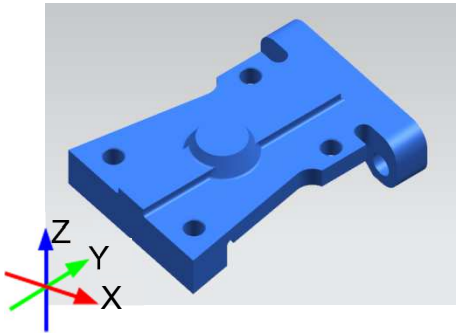
4.9.2 Level-2 (L-2) Genetic Algorithm (GA)

At this level of optimization, analysis is performed on the set of Tool Diameters (TD) selected for minimizing Non-Machinability of the part. This allows for optimization per orientation by iterating and evaluating through multiple tool configurations. This allows for minimizing Non-Machinability with optimized TD for each depth for every orientation in a given set. The details for the L-2 GA including parameters and corresponding operations are shown in figure 32.

$$\begin{matrix} \\ \begin{bmatrix} 0 \\ 90 \\ 180 \\ 270 \end{bmatrix} \end{matrix} = \begin{matrix} TD_0, & TD_1 \\ \begin{bmatrix} 0.500, & 0.125 \\ 0.250, & 0.187 \\ 0.375, & 0.125 \\ 0.500, & 0.187 \end{bmatrix} \end{matrix}$$

Chromosome = (0.500, 0.125, 0.250, 0.187, 0.375, 0.125, 0.500,

Figure 32: Value encoding



	Axes	Axis	Theta	$Depth_{max}$	Tool diameter
1	1	X	0	0.525	0.250
		X	90	0.575	0.125
		X	270	0.670	0.125
Dimensions (inch)				Triangles	Time (mins)
2.75 X 3.75 X 1.15				26076	539.2

Figure 33: Example part for GA

The encoding used in the L-2 GA is value-based, where every set in the seed population contains the TD value in inches (Figure 33). The input to GA at L-2 is a population of set of tool diameters and the candidate set for %NM for each setup can be evaluated. This encoding is further used on which all other GA operations are used. The

crossover operator used on this chromosome is a 2 site crossover where the values between a set of parents are exchanged. In mutation, a site in the chromosome is chosen as random and the value in it changed to one chosen randomly from the generic tool diameter library designed by the author. The selection method used in L-2 is *Elitism*, where the best candidates are always stored and used in the next generation with the hope to converge on an optimal set of tool diameters. The convergence and the results from GA used in both L-1 and 2 are shown below for an example part (Figure 34).

4.9.3 Example of GA implementation

In order to test the results of the chosen GA schedule, an example prismatic part (Figure 34) is shown using setups determined using the fitness function and the selected GA schedule (Table 2). The convergence criteria selected for both levels was to complete the number of allotted generations using the GA parameters chosen. In GA, a better criterion could be to run the procedure until there is no significant change in the

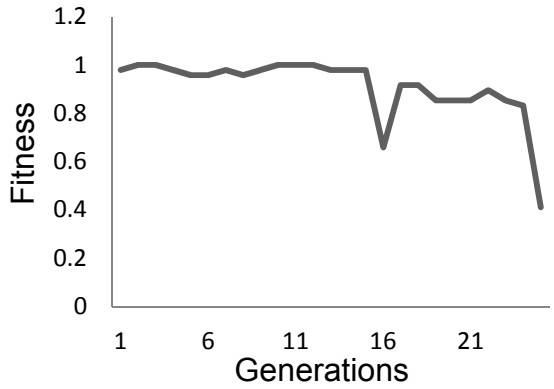


Figure 34: L-1 GA convergence

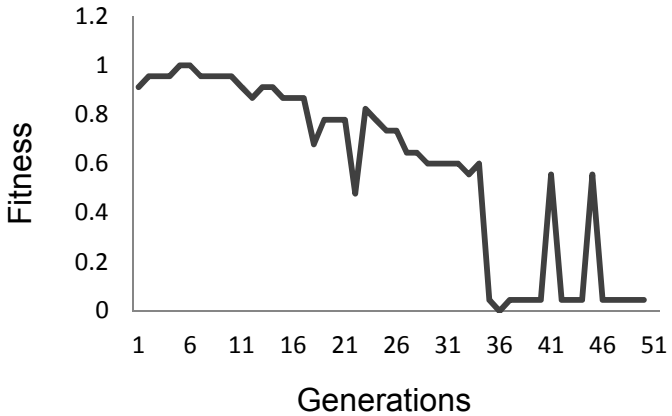


Figure 35: L-2 GA convergence

solution for a selected number of generations. However in the current implementation this criterion was not chosen due to the fact that the slicing procedure implemented on the CPU consumes a huge amount of time that would result in the GA running for many hours.

Level-2 convergence

Figure 35 shows the convergence attained at L-2 where C-2 (Table 2) is minimized using multiple tool diameters. The fitness function improved upon is the Non-machinability component. In this case, due to the prismatic nature of the part, it is possible to determine the setups within the allotted number of generations. However when complicated part geometries are involved, the number of generations and population required might increase to achieve convergence to a good set of orientations and tool configurations.

Level-1 convergence

Figure 36 shows the convergence attained for the entire fitness function that would provide a goodness measure of a given set of setups. The values of all three components (C1, 2, 3) are plugged in the fitness function to provide the goodness measure of a given set of setup orientations.

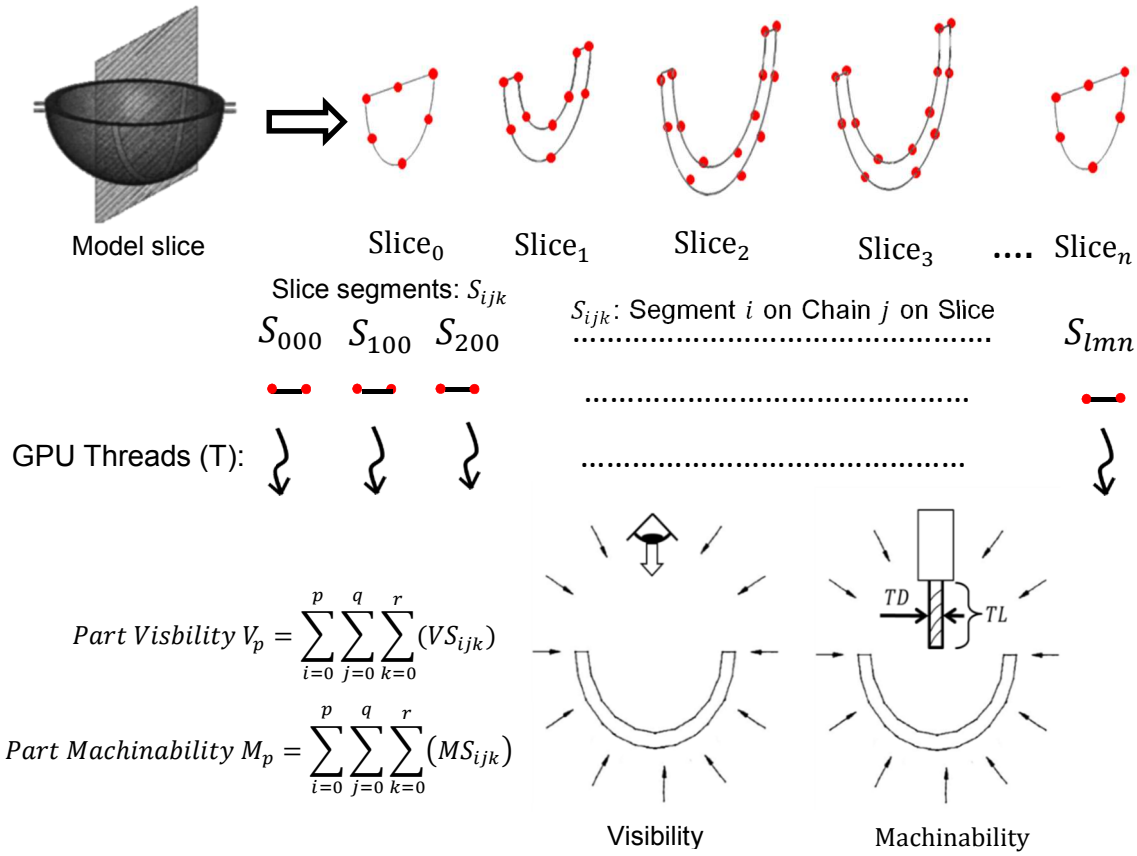


Figure 36: GPU implementation: Visibility, Machinability and Depth analysis using slice model

4.10 Parallel Processing For Implementation

The problem of setup planning presented in this paperwork gains results from visibility [4], machinability [9] and depth analysis algorithms using a slice model as input. However these algorithms for determining setups are (n^2) in nature which results in a significantly longer time to get results. On the other hand, the use of a 2½-D slice model provides an opportunity to analyze all segments in parallel (Figure 37). This problem comes under the *embarrassingly parallel* category where there is little to no dependency between parallel tasks. Thus, in order to take advantage of this situation, we used parallel processing with a GPU.

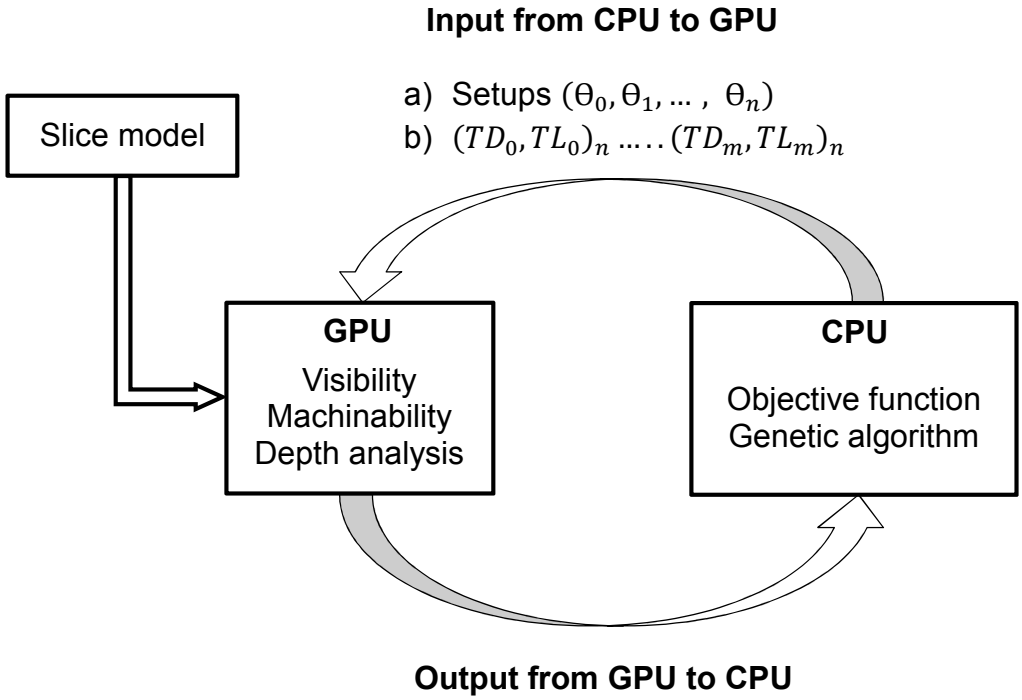


Figure 37: Hardware structure for implementation

In this paperwork, a combination of CPU and GPU computing is deployed (Figure 38). Specifically the GPU was used for analyzing the parameters %NV, %NM and machining depths using slice model input from a given set of orientations. The results obtained from the GPU were plugged into the objective function implemented on the CPU.

4.11 Results

The desktop used for implementation was a 3.6 GHZ, Intel Xeon processor with 16GB RAM. For parallel processing an NVIDIA GPU C-2075 with a compute capability of 2.0 using CUDA-C language was used. The implementation on CPU was done using C++ using Visual Studio 2012 using openGL glui interface. In order to verify the results,

examples of two polygonal models, one prismatic and one freeform are shown below. A tool library is designed and used in the optimization routine to minimize the percent non-machinability (table 4).

Table 3: Model dimension (inch)

Model	X	Y	Z	Triangles
Prismatic	4.40	2.00	1.50	28541
Freeform	5.99	1.27	1.28	41776

Table 4: Tool library

Sr No	Diameter (inch)	Length (inch)
1.	0.5000	(1.0 – 5.0)
2.	0.3750	
3.	0.2500	
4.	0.1875	
5.	0.1250	

Example Model 1: Prismatic Model

Results

For the prismatic model shown in the figure 39, the setup orientations and the corresponding tool diameters chosen through the optimization and genetic algorithm procedure are shown in table 5. As shown, the setup results for this model were about multiple axes.

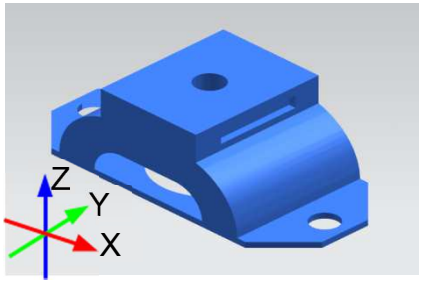


Figure 38 Prismatic model

Table 5: Prismatic model results

	Axes	Axis	Theta	$Depth_{max}$	Tool Diameter	
1	2	**X**	**0**	1.1698	0.1875	0.1875
		X	**180**	1.0697	0.500	0.1875
		X	**270**	1.1557	0.1875	0.1875
		Y	**90**	1.6669	-	0.1875
		Y	**270**	1.6669	-	0.1875
Time (mins)				% Non-machinability		
461				92.2		

Verification

For verifying the non-machinability results, tool paths were generated in MastercamX6 using the determined setup orientations and tool diameters. Figures 41 and 42.b show the model geometry created using tool path simulation where the non-machinable areas exist inside the oval shape or arrows. This can be compared to the red regions depicting non-machinable areas in Figures 41 and 41.a, as predicted by the methods of this work.

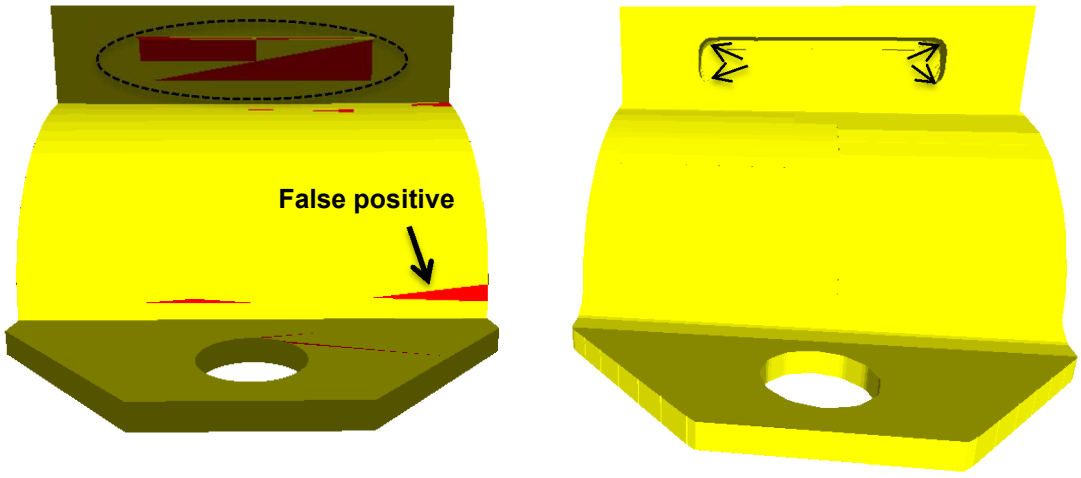


Figure 39 (a) Predicted (b) Simulated non-machinable regions (arrows)

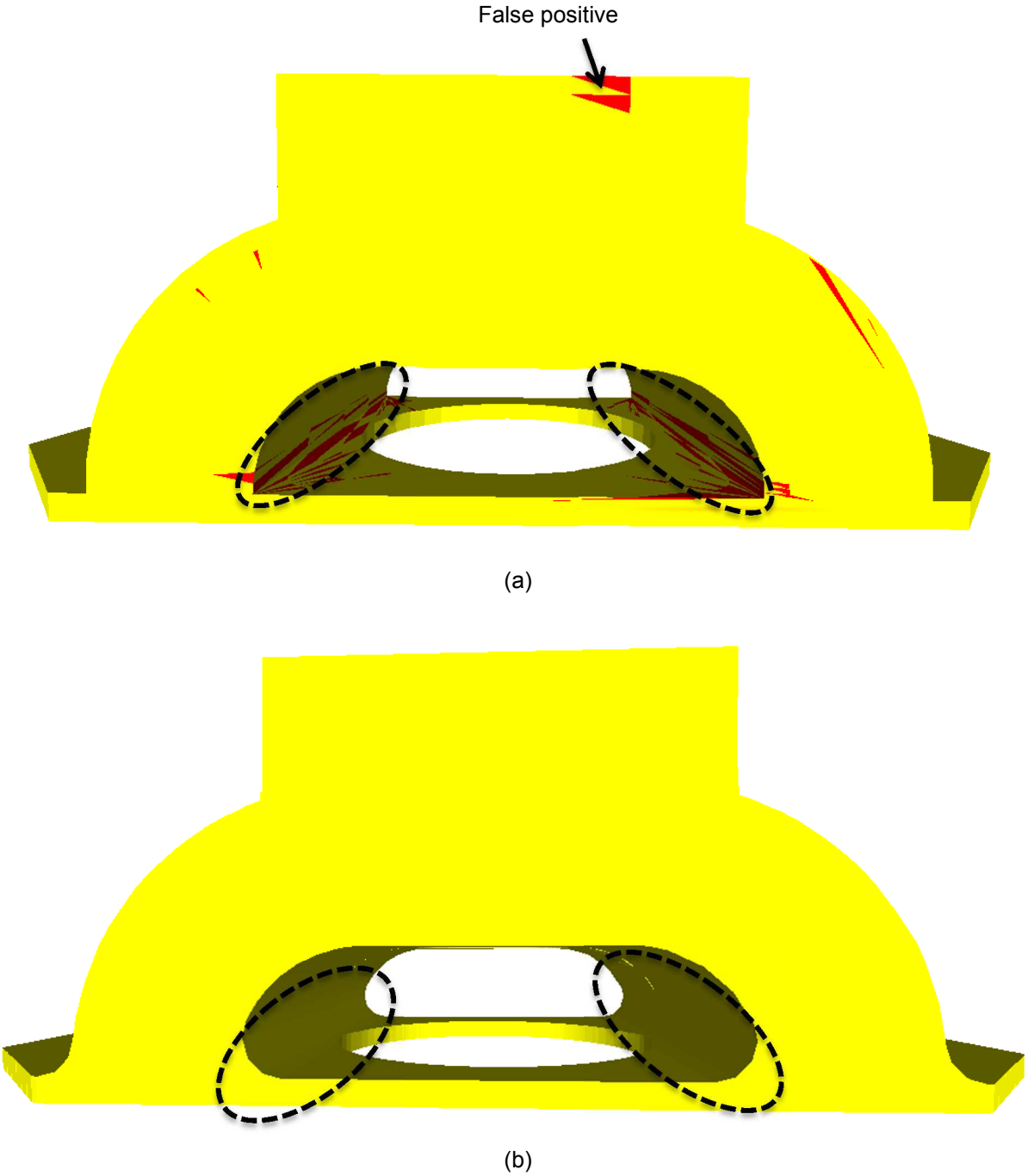


Figure 40 (a) Predicted (b) Simulated non-machinable regions (inside dotted polygon)

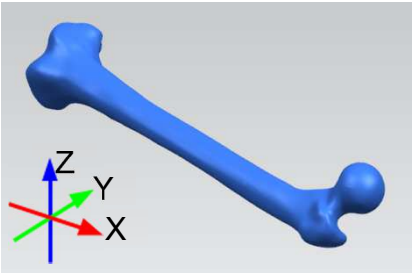


Figure 41: Freeform model

Table 6: Freeform model results

	Axes	Axis	Theta	$Depth_{max}$	Tool Diameter
2	1	X	10	0.640	0.375
		X	85	0.693	0.375
		X	190	0.872	0.125
		X	265	0.780	0.125
Time (mins)				% Non-machinability	
372				98.9	

Example Model 2: Freeform Model

Result

Next, a freeform femur bone model shown in the figure 43 is used to determine the setup orientations and the corresponding tool diameters shown in table 6. In this case the setups solution determined were about a single axis.

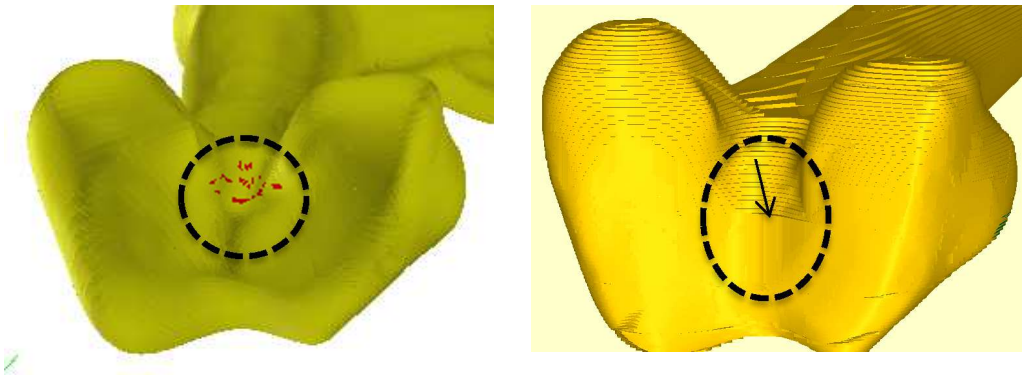


Figure 42: (a) Predicted (b) Simulated non-machinable regions

Verification

Tool paths were simulated in MastercamX6 using the determined setup orientations and tool diameters, where Figure 43.b shows the model geometry created by the tool path simulation and the non-machinable areas are shown inside the oval. This can be compared with the red regions depicting non-machinable areas in the figure 43.a, as predicted by the methods of this work.

4.12 Limitations In Implementation

Some of the important issues faced in this work are the need for a better criterion for GA convergence and false positives and false negatives in the non-machinability results.

GA convergence

From the setup solution in the results section, it can be seen that the time taken to achieve the setup solutions for models have been in hours. This has been largely due to the slicing procedure implemented on the CPU. Slicing time is determined by the number of slices to be generated, which depends on the model dimensions as well as number of triangles (Table 7). The slice spacing used in this paperwork is rather small (0.02 inches), in order to ensure that triangles do not remain. One solution to reduce the analysis time is to port the CPU slicing algorithms to the GPU, but that work has not been completed.

Table 7: Slicing time (0.02 slice spacing)

Model	Triangles	Dimension	Slices	Slicing time (secs)
1	26076	2.75 X 3.75 X 1.15	137, 187, 57	4.67, 5.23, 1.1
2	1938	3.20 X 0.70 X 0.64	160, 35, 32	4.87, 1.0, 1.0,
3	45010	6.59 X 5.34 X 3.21	329, 267, 160	6.92, 5.92, 4.19

False positive/negative non-machinability results

Figure 44 shows the non-machinable regions (red) on the model predicted by the setup solutions, where some obvious false positives or negatives are present. This has been largely due to geometric issues such as gaps or saw tooth shapes, and overlapping segments in the slice file inherited from STL models. Another error source is floating point precision, where the number is rounded off by the CPU and cause a segment and its corresponding triangle to be tagged as machinable not in some cases. Both of these problems can been tackled by using geometrically sound slice models as well designing code that handles floating point precision soundly in future implementations.

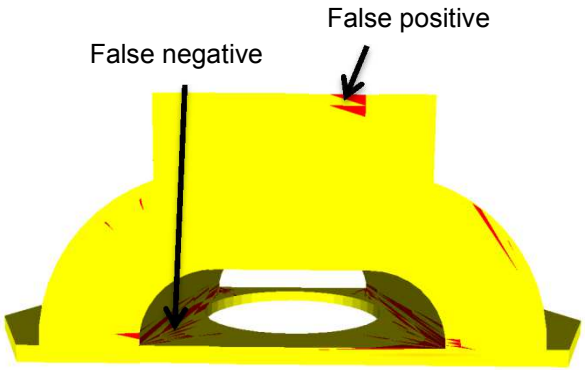


Figure 43: Predicted non-machinable regions (red)

4.12 Order Of Algorithms

It can be seen from the results that the overall run time to get setup orientations is significantly high. This can be attributed to the run time required to get process planning results used in the implementation. The implementation was done partly on GPU and partly on CPU where the usage of GPU helped reduced the runtime to analysis results significantly. Out of all the algorithms used, the visibility $O(n^2)$, machinability $O(n^2)$, and reachability $O(n^2)$ were implemented on GPU where each GPU thread was allotted for analysis of individual point on the slice model effectively reduced the runtime to $O(n)$. Hence in this area, the use of GPU significantly helped reduced the analysis time.

However the STL slicing algorithm $O(n^2)$ and multi-level GA [O(gnm)*O(gnm)] were restricted to CPU only, where g is generations, n is population size and m is number of individuals in a set in GA. This shows that the significantly high run time is mainly due to the multi-level GA on CPU. The GPU device used in implementation has only one level of parallelism allowed due to which the GA had to be restricted to CPU and hence significantly higher run time. The next step would be to use a GPU device that allows for nested parallelism which would enable use of GA in parallel and reduce the overall runtime to that of slicing algorithm $O(n^2)$. Additionally the usage of parallel slicing would effectively reduce the overall runtime of the implementation to O(n). This would be of significant impact considering that runtime of O(n) due to use of GPU allowing extensive analysis of design space would provide a near optimal machining orientations as compared to $O(n^2)$ on CPU that always provided sub-optimal solutions using primary axes.

4.13 Conclusions And Future Work

The methods presented in this paper have successfully provided an automated system for determining setup orientations for creating prismatic and freeform models using discrete 3-axis machining. Some of the future work considerations include improvements in the objective function, where the quality of additional variables like fixturing and tolerancing could drive towards better setup solutions. Additionally, this method focused on 3 primary axes for setup planning with assumptions that prismatic parts are generally accessible about these axes. However, any number of off-axes could be considered for setup planning using these methods. These algorithms could also be used in an automated *manufacturability* analysis system. In this manner, the

part model could be analyzed about multiple axes and the analysis results could be provided in the form of feedback to the designer about the part's relative complexity and ease of manufacture.

4.14 References

[1] Woo, Tony C. "Visibility maps and spherical algorithms." *Computer-Aided Design* 26.1 (1994): 6-16.

[2] Chen, Lin-Lin, and Tony C. Woo. "Computational geometry on the sphere with application to automated machining." *Journal of Mechanical Design* 114.2 (1992): 288-295.

[3] Tang, Kai, J. Gan, and Tony Woo. Maximum intersection of spherical polygons and workpiece orientation for 4-and 5-axis machining. (1992).

[4] Frank, Matthew C., Richard A. Wysk, and Sanjay B. Joshi. "Determining setup orientations from the visibility of slice geometry for rapid computer numerically controlled machining." *Journal of manufacturing science and engineering*128.1 (2006): 228-238.

[5] Liu, Zhenkai, and Lihui Wang. "Sequencing of interacting prismatic machining features for process planning." *Computers in Industry* 58.4 (2007): 295-303.

[6] Chu, Chi-Cheng Peter, and Rajit Gadh. "Feature-based approach for set-up minimization of process design from product design." *Computer-Aided Design*28.5 (1996): 321-332.

[7] Sarma, Sanjay E., and Paul K. Wright. "Algorithms for the minimization of setups and tool changes in "simply fixturable" components in milling." *Journal of Manufacturing Systems* 15.2 (1996): 95-112.

[8] Balasubramaniam, Mahadevan, et al. "Tool selection in three-axis rough machining." *International Journal of Production Research* 39.18 (2001): 4215-4238.

[9] Li, Ye, and Matthew C. Frank. "Machinability analysis for 3-axis flat end milling." *Journal of manufacturing science and engineering* 128.2 (2006): 454-464.

[10] Fountas, Nikolaos, et al. "Single and multi-objective optimization methodologies in CNC machining." *Statistical and Computational Techniques in Manufacturing*. Springer Berlin Heidelberg, 2012. 187-218.

[11] Nallakumarasamy, G., et al. "Optimization of operation sequencing in CAPP using simulated annealing technique (SAT)." *The International Journal of Advanced Manufacturing Technology* 54.5-8 (2011): 721-728.

[12] Liu, Xiao-jun, Hong Yi, and Zhong-hua Ni. "Application of ant colony optimization algorithm in process planning optimization." *Journal of Intelligent Manufacturing* 24.1 (2013): 1-13.

[13] Lim, W. C. E., G. Kanagaraj, and S. G. Ponnambalam. "A hybrid cuckoo search-genetic algorithm for hole-making sequence optimization." *Journal of Intelligent Manufacturing* (2014): 1-13.

[14] Zuperl, Uros, and Franci Cus. "Neural Network Assisted Particle Swarm Optimization of Machining Process." *Key Engineering Materials* 581 (2014): 511-516.

[15] Corso, L. L., et al. "Using optimization procedures to minimize machining time while maintaining surface quality." *The International Journal of Advanced Manufacturing Technology* 65.9-12 (2013): 1659-1667.

[16] Zhong, Pei Si, et al. "Method of Holes Machining Path Optimization for NC Turret Punch Press." *Advanced Materials Research* 842 (2014): 553-557.

[17] YIP-HOI, DEREK, and DEBASISH DUTTA. "A genetic algorithm application for sequencing operations in process planning for parallel machining." *IIE transactions* 28.1 (1996): 55-68.

[18] Li, W. D., S. K. Ong, and A. Y. C. Nee. "Hybrid genetic algorithm and simulated annealing approach for the optimization of process plans for prismatic parts." *International journal of production research* 40.8 (2002): 1899-1922.

[19] Khan, Z., B. Prasad, and T. Singh. "Machining condition optimization by genetic algorithms and simulated annealing." *Computers & Operations Research* 24.7 (1997): 647-657.

CHAPTER 5: ADDRESSING PROBLEMS WITH GEOMETRIC SINGULARITIES UNIQUE TO 2D SLICE MODELS

5.1 Introduction

In the field of Computer Aided Process Planning (CAPP) for manufacturing, a variety of CAD formats have been used for analysis in order to make components for different industries. Two common file *types* have typically been used for CAPP; (a) Feature-based, where individual features on the model geometry can be identified (Figure 1), and (b) Feature-free, where the surface geometry is approximated by polygons or parametric surfaces (Figure 2). Parametric models are widely used for CAPP and for the generation of tool paths for processes such as CNC machining.

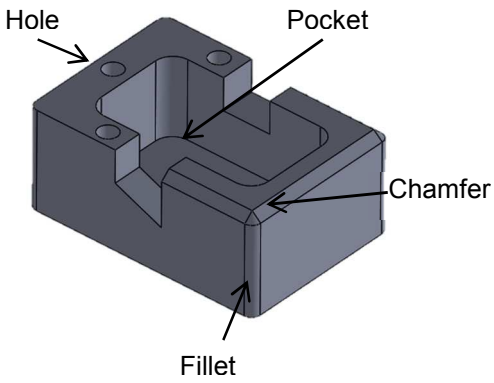
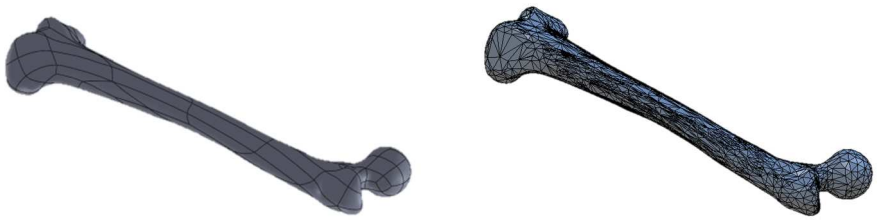


Figure 1: Feature based CAD

However when it comes to processes like additive manufacturing, they generally use slice models created from standard polygonal STL models (Figure 3). These slice models are a stack of 2D cross sections formed by slicing each triangle on the STL model by a plane and then converting to 2D chains (Figure 4). However triangles parallel to the slice plane are always missed (no line intersection) and therefore data on the STL model is missed. This limitation does not affect the 3D printed part geometry, since they only process what is sliced and then output layers of material based on two slices. That is, any plane that falls between slices will simply appear as the flat surface of an adjacent layer surface anyway. However it becomes important to capture the complete model geometry when multi-surface models are involved or multi-axis

manufacturing processes are used. Hence this chapter provides a two-step solution to dealing with flat planes that are parallel to the slice direction. The first step is to develop a new method that enables slicing of all triangles on the polygonal models giving a new *hybrid* slice model. The second step involves a new method that enables CAPP for multi-axis rapid machining using the hybrid slice model created in the first step.



(a) (b)

Figure 2: Feature free CAD (a) Parametric (b) Polygonal

5.2 CAPP For Additive Rapid Manufacturing

Additive Manufacturing (AM) processes like Stereolithography (SLA), Electron Beam Melting (EBM), and Selective Laser Sintering (SLS) etc. have mainly used polygonal models to create 2½-D slice models and to perform automated process planning(Figure 3). Slice models consist of a stack of 2-D cross sections created by slicing the polygonal models along an axis (Figure 4). Every 2-D Slice (S) consists of a single or multiple closed polygon chains (C) where each chain consists of an ordered set of segments (Seg). Hence, each segment on the slice model has a unique index such that a segment $Seg_{i,j,k}$ would be an i^{th} segment, on the j^{th} chain, of the k^{th} slice; *i, j, k* being whole numbers. Segments on the slice model chains are the result of intersection between slice planes and connected triangles that result in the polygonal chains. Each

triangle can be sliced multiple times such that multiple segments are created from the same triangle (Figure 3). In this way the slice model approximates the geometry of the polygonal model.

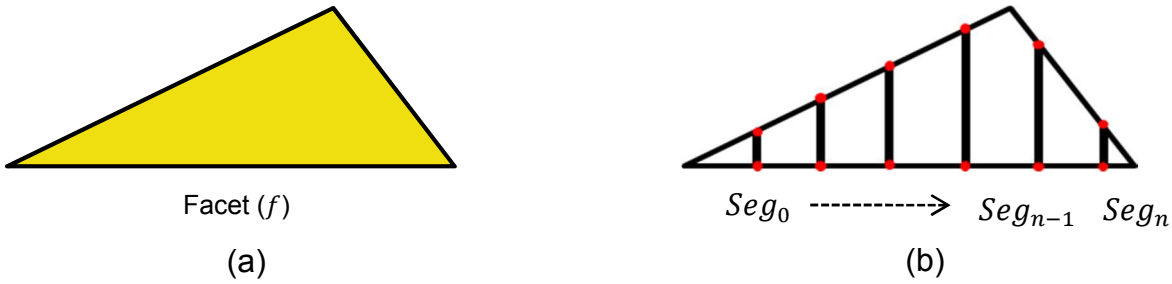


Figure 3: (a) Triangular facet, (b) Segments on sliced facet

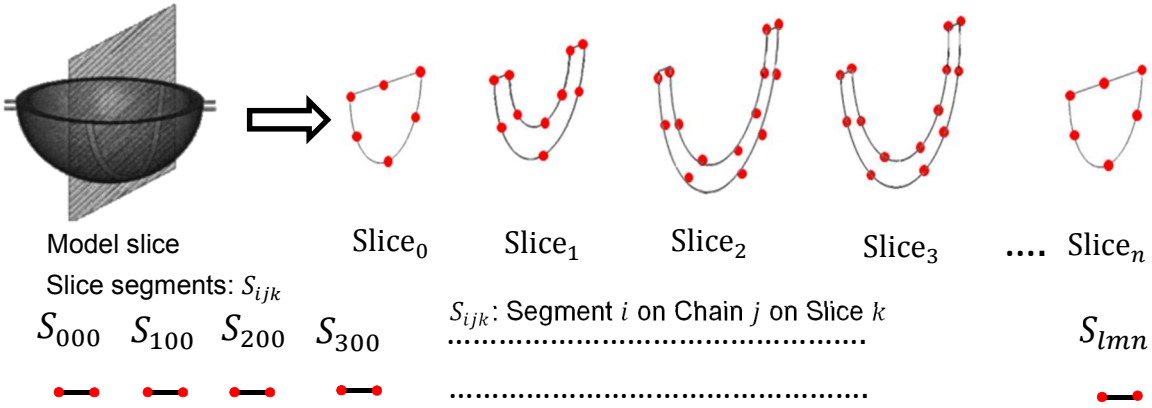


Figure 4: (a) Polygonal model (b) Slice model

The level of geometric approximation depends on the number of slices on the model, (i.e. more slices gives a better approximation). Slice models used in process planning for additive manufacturing enable considerations on build directions, support generation, material deposition paths, etc.

5.3 CAPP For Subtractive Rapid Manufacturing

Algorithms that use slice models for subtractive rapid manufacturing (Figure 5) were developed by Frank et al [19]. In that research, algorithms were developed for

determining the visibility of a slice model for setup planning. The line-of-sight visibility for each segment was computed about a basis of 0°-360° and then a set of machining orientations from which all the segments are visible could be determined. The visibility for a given segment in these algorithms is categorized into 2 types, a) local visibility i.e. the segment visibility with respect to the chain it belongs to on the same slice (Figure 6.a) and b) global visibility i.e. the segment visibility with respect to all the chains excluding the chain it belongs to on the same slice (Figure 6.b). Hence using the visibility information of the segments on the slice model, and using a meta-heuristic approach, the minimum number of orientations required for machining the part about the given rotary axis is determined. This method is restricted to determining visibility and orientations for machining models only about a specific rotary axis (Figure 5). However there could always be complex part designs that are not visible completely about any one rotary axis. In this case there would be a need for developing algorithms for CAPP that would allow machining of complex part designs about multiple axes. This in addition to a single rotary axis would require CAPP on the part models about multiple axes including analysis but not limited to visibility, accessibility, and fixture planning, etc.

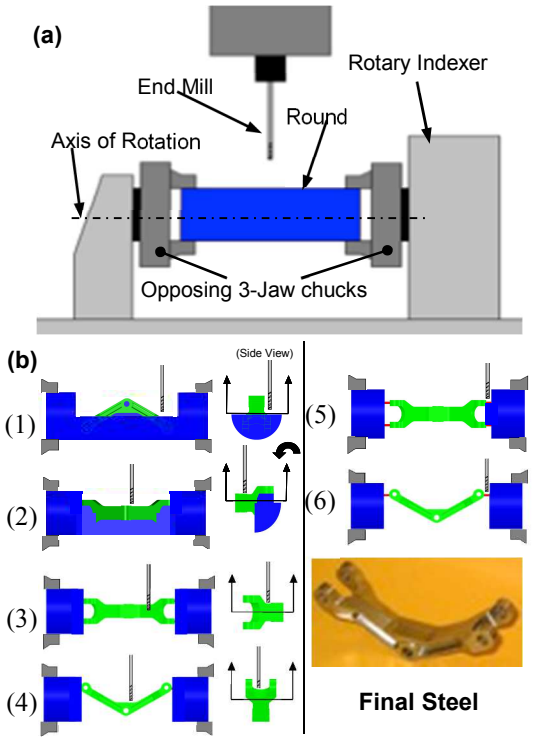
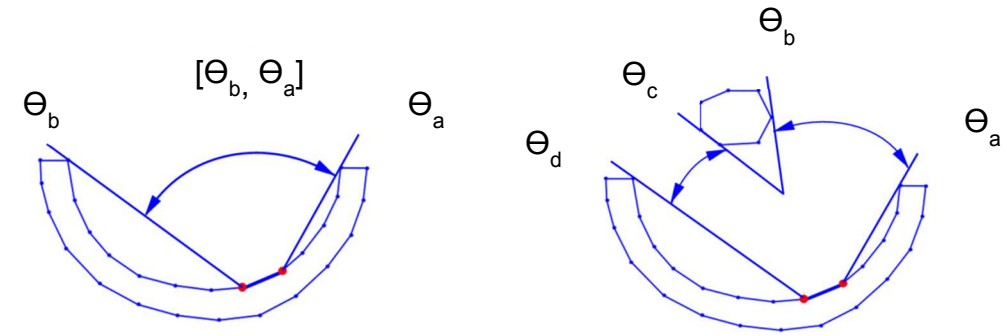


Figure 5 - (a) CNC-RP setup; (b) steps b.1-b.4 expose component geometry while b.5-b.6 exposes sacrificial supports

Chapter 4 presented a new CAPP method that enables automated setup planning for making parts using multi-axis CNC machining process.



(b) Segment visibility [Θ_a, Θ_b] (c) Segment visibility [Θ_a, Θ_b], [Θ_c, Θ_d]

Figure 6: Segment visibility a) Local b) Global

5.4 Combined Use Of Polygonal And Slice Models For CAPP

In chapter 4 the author presented a new CAPP method for determining setup orientations for discrete 3-axis machining process using polygonal models. Using the visibility method developed by Frank et al [19], the slice models generated about each axis were analyzed. Since the application of slice models were only about a given axis, a modified facet-based visibility was introduced which allowed determination of every facet about all three axes. In this method the visibility information on each slice model about each axis is mapped back to the polygonal model. This provides the visibility results for the regions that would be visible or otherwise given multiple combinations of 3-axes, thus giving the combined visibility information of the polygonal model (Figure 7).

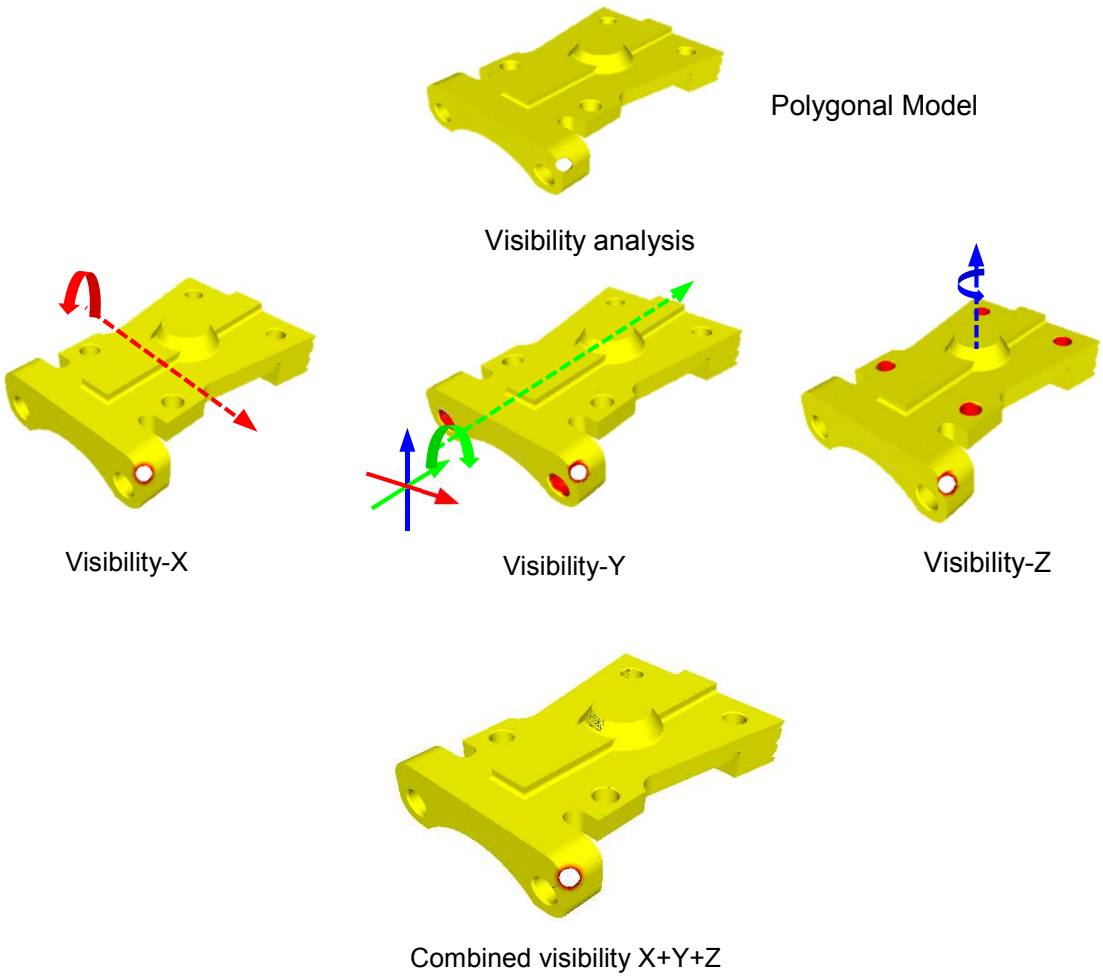
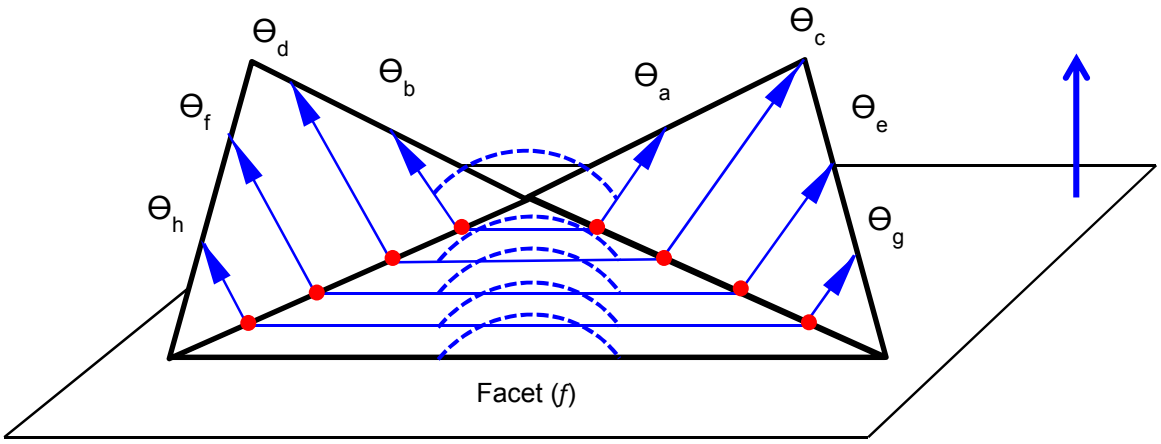


Figure 7: Model visibility

5.4.1 Visibility mapping from slice to polygonal model

In order to determine the visibility of the polygonal model as presented above, it is necessary to determine the visibility of each facet. This was done by mapping back the visibility from each slice model about a given axis to the polygonal model. In this method the segments on the slice model were grouped depending on the triangle from which they were created (Figure 8). The visibility of segments in each group was combined to determine the triangle visibility about a specific axis (Figure 8). In this manner the

visibility of all the triangles present on the polygonal model could be computed about



Segment visibility $Sg_{i,j,k}$: $[\Theta_a, \Theta_b]$, $Sg_{i,j,k+1}$: $[\Theta_c, \Theta_d]$, $Sg_{i,j,k+2}$: $[\Theta_e, \Theta_f]$, $Sg_{i,j,k+3}$: $[\Theta_g, \Theta_h]$

Facet (*f*) visibility: $[\Theta_a, \Theta_b] \cap [\Theta_c, \Theta_d] \cap [\Theta_e, \Theta_f] \cap [\Theta_g, \Theta_h] = [\Theta_a, \Theta_b]$

Figure 8: Facet (*f*) visibility

any given axis by the mapping method, allowing determination of the combined visibility percentage of the polygonal model about all the 3-axes. Following this it would be possible to determine an optimal set of machining orientations that would allow machining of the polygonal model about the 3-primary axes.

5.5 Challenges In Use Of Polygonal And Slice Model

Polygonal models have other geometric issues like gap, overlapping polygons (faces), non-manifold topology, inverted polygon normal, degenerate polygons or intersected polygons. Significant research has been done to address geometric challenges related to polygonal models in order to manufacture parts with better geometric accuracy. The geometric issues in polygonal models propagate to faulty slices that may lead to significant errors in the produced parts. One among other challenges while generating slice models is the inability to capture complete surface information from the polygonal model due to coarse slice spacing (Figure 9). There have been methods in the research

community which propose the use of adaptive or similar intelligent slicing procedures. These have been successful in capturing surface information from the polygonal models

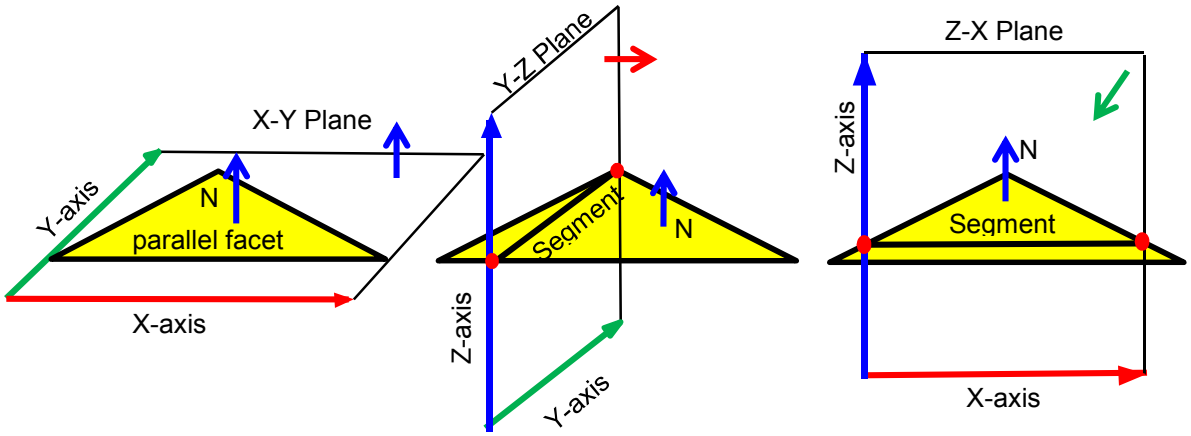


Figure 9: Facets (a) Non-sliceable (b) Sliceable (c) Sliceable

to a great extent. Significant research performed in addressing geometric challenges related to polygonal models is summarized in the literature review section.

5.6 CAPP Challenges Using Slice Model For Multi-axis Machining

One of the limitations encountered in the use of slice models for multi-axis CAPP is a unique inability to slice the facets that are parallel to slicing plane. Slice models for CAPP for subtractive rapid manufacturing about an axis have been extremely valuable towards machining accurate parts. However in this method to determine an optimal setup solution, it is deemed necessary that visibility to every facet must be determined about an axis to get the total model visibility about an axis. In this way when model is analyzed about multiple axes, the determined visibility of all the triangles about all the axes allows a correct setup decision. This revealed a geometric limitation in the use facet visibility technique. The unique condition discovered was the inability to slice the facets parallel to the slice plane (Figure 10). This would create slice models with surface

regions missing which were parallel to slice plane. In this case it would be impossible to determine the visibility of triangles parallel to slice plane since no segments would be created from them during slicing. This makes it impossible to analysis the complete model visibility facet wise about any given axis. CAPP due to this is affected in two major areas of CAPP 1) CAPP for discrete 3-axis process using polygonal models

.

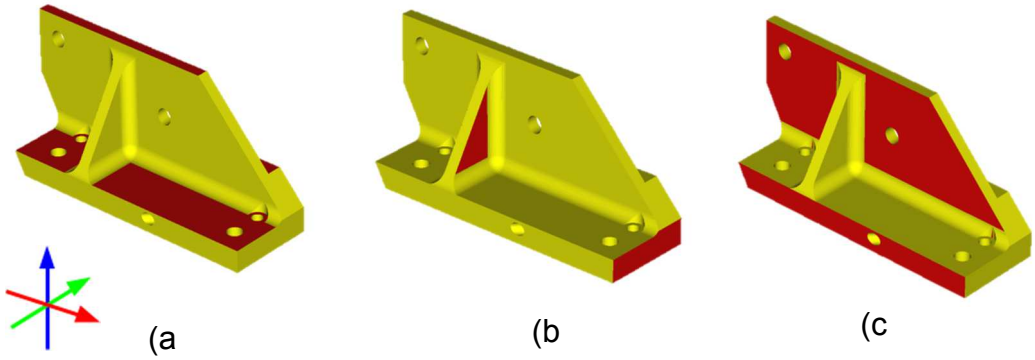


Figure 10: Non-sliceable facets (red) parallel to (a) X-Y plane (b) Y-Z plane (c) Z-X

Table 1: Sliceable, Non-sliceable area

Axis	Total area (in^2)	Sliceable area (in^2)	Non-sliceable area (Parallel area) (in^2)
X	134.9	91.3	43.6
Y		129.3	5.6
Z		78.1	56.8

5.7 CAPP For Multi-axis Machining

5.7.1 CAPP for discrete 3-axis process using polygonal models

In the section above it was explained that the facet based visibility method gives rise to geometric issue specific to non-slicing of triangles parallel to the slicing plane (Figure 12). This could exist for any type of polygonal models, whether prismatic or freeform in nature.

5.7.2 CAPP using multi-colored polygonal models

In addition to standard polygonal STL format, colored polygonal formats such as PLY, VRML, and OBJ etc. are also used for additive and subtractive manufacturing. In chapter 3 new CAPP methods to machine multi-colored models about a rotary axis with specific application to multi-surface bone implants were presented. The colors stored in the polygonal format could represent various attributes such as tolerance, surface finish, hardness, and texture etc. Hence it is necessary to pass these attributes from triangles to the segments originating from them such that CAPP can consider them. The issue of non-sliceable facets is compounded if multi-axis CAPP is performed on multi-colored models. This work presents a unique and simple two-step solution to the problem; 1) Bringing parallel facets into consideration, and then 2) providing modified visibility algorithms such that the visibility of segments created from parallel facets will be included.

5.8 Literature Review

Since the invention of Rapid Prototyping (RP) systems, extensive research has been undertaken on addressing geometric problems of polygonal and corresponding slice models to maintain part quality. Several problems related specifically to STL file

formats were studied, for example; gaps, degenerate facets, overlapping facets and non-manifold topology conditions have been considered [1, 2]. In [1], the authors specifically addressed the problem of missing facets while [2] addressed the problems such as multiple gaps at coincidental vertices. In [3] a method was presented to detect defects in the surface representation and to analyze the shape of the approximated surface by constructing a polyhedral data structure from an STL file. [4] presented a substantial survey on polygonal model geometric errors, their sources and possible techniques for fixing them. In [5] these geometric polygonal problems were considered as a mesh boundary decimation task where a new vertex-edge collapse operation was introduced. This provided extra supports for closing gaps and stitching together the boundaries of triangle patches lying in near proximity to each other. [6] presented a robust method for repairing arbitrary polygon models by using an Octree grid and reconstructing the surface by contouring. This partitioned the model space into disjoint internal and external volumes. In [7] a user friendly interactive graphical tool was introduced which incorporated several mesh repairing features and allowed a low-level editing which was claimed to be missing in most other existing software packages. This application was provided as a post processing tool for scanned surface models. [8] provided a fully automated solution that converted an inconsistent input mesh into an output mesh that was guaranteed to be a clean and consistent mesh representing the closed manifold surface of a water tight model. [9] presented a light weight solution that converted a low-quality digitized polygon mesh to a single manifold and watertight triangle mesh without degenerate and intersecting elements. There has been significant research performed on addressing challenges related to the slice models generated

from different polygonal formats. These slice models have been regularly used as input to process planning systems for Additive Manufacturing (AM) processes. The geometric accuracy of the slice model directly affects the quality of the part made from RP systems. One of the reasons the geometry and accuracy related challenges arise on slice models is due to the issues on the corresponding polygonal models. In [10] work was presented where the slice model generated from a polygonal model had chains with gaps and over-lapping segments. These contour anomalies were processed in the 2D slice model to obtain error free slices suitable for process planning. [11] Proposed a tolerant slicing algorithm for processing slice contours for AM systems that also addressed memory concerns for storing slice information. The algorithm provided a maximum bound tolerance that directly related to the gap between two segments and allowed them to close the gap if it was within the tolerance. [11] Also discussed rare issues like single/multiple unbound contours, surplus line segments that may arise on slice models. Another challenge is addressing geometric accuracy, build time; computational memory concerns in RP produced parts, which are the result of slice spacing choice. Significant research has been done on the topic of automated slicing strategies that allow the process planning system to control the part build time, optimize material use and memory as well as maintain the required geometric accuracy. [12] Presented a new adaptive slicing method that used both uniform as well as selective slicing to maintain optimum build time and high surface accuracy. There have been numerous other work on process specific slicing strategies for maintaining surface finish, geometric accuracy, and build speed [13][14][15][16][17][18]. In summary, significant research has been done on polygonal models and corresponding slice

models that have addressed challenges related to geometric accuracy, surface finish requirements, part production time, material usage and computational memory requirements of the part to be produced.

5.9 Overview Of Solution Method

This work proposes to first create a standard slice model, while detecting parallel facets about the axis about which the standard slice model is created. These parallel polygons are then sliced using an arbitrary slice plane to give individual segments parallel to the primary slice plane. These new parallel segments are then merged with the standard slice model to give a hybrid slice model (Figure 13). The new hybrid slice model thus has complete surface data from the polygonal model and can be used for process planning for multi-axis machining (Figure 14).

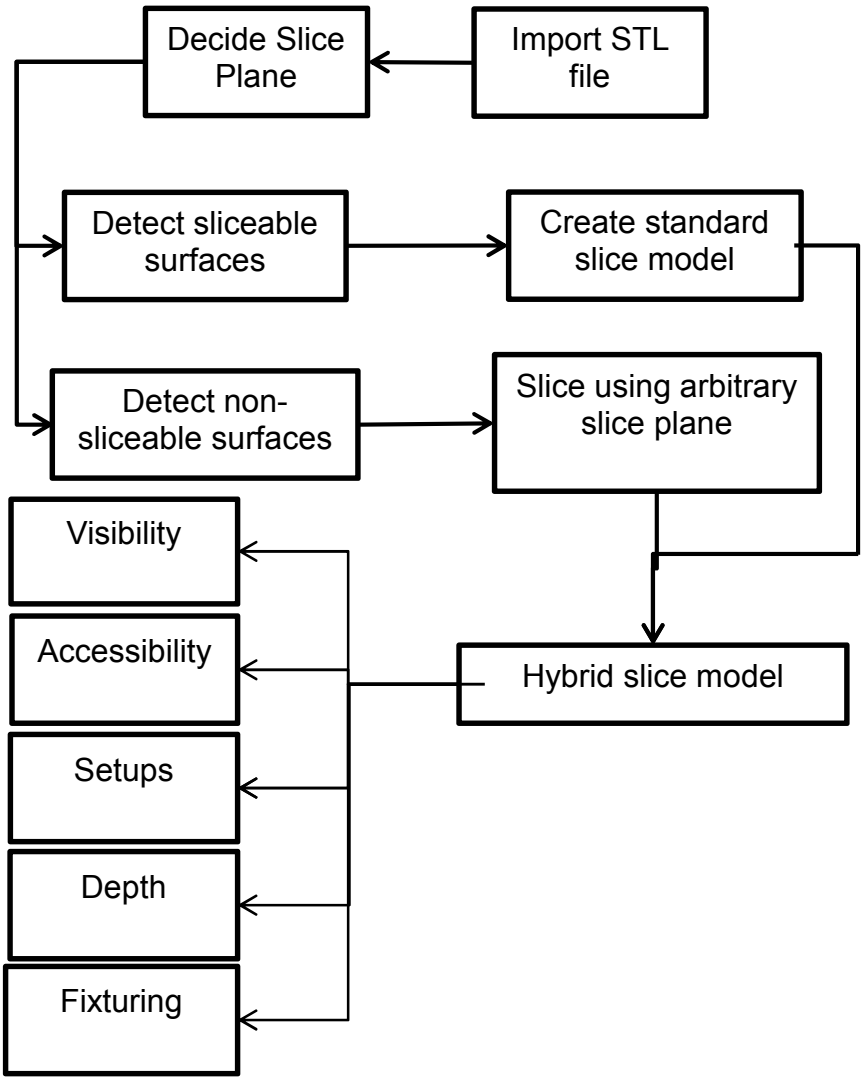


Figure 11: Proposed approach

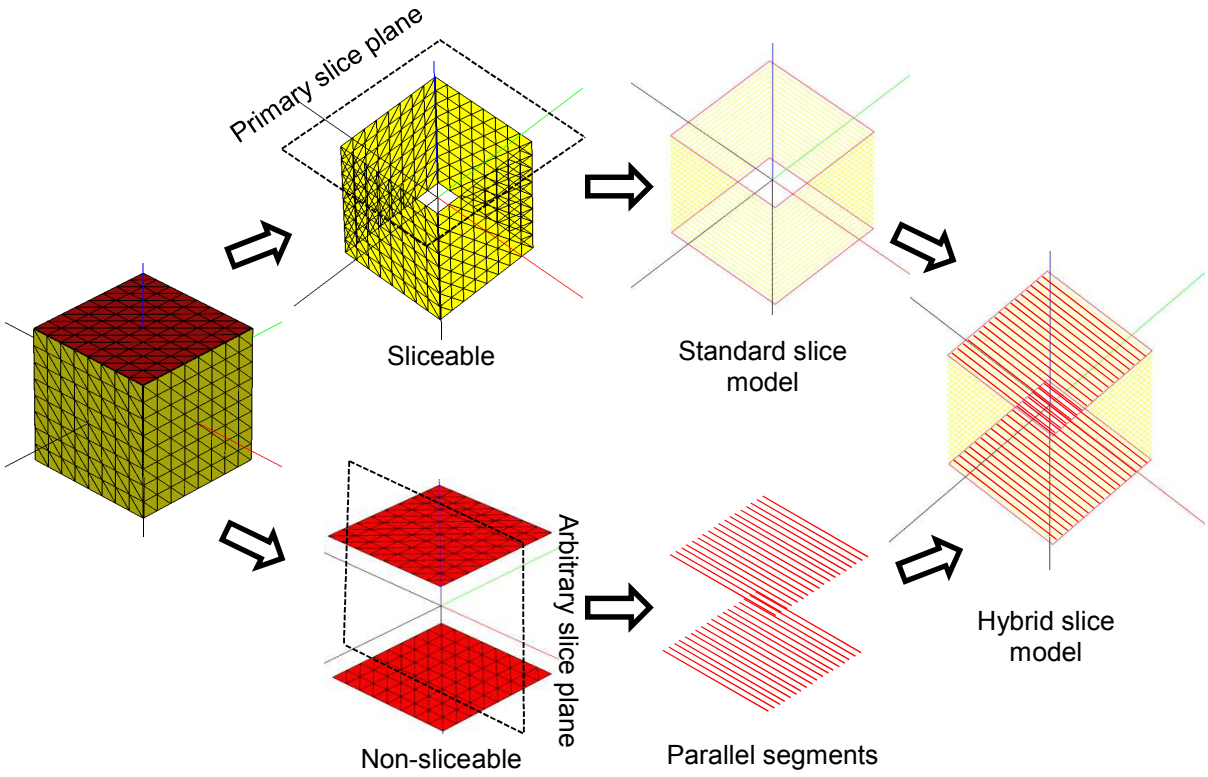


Figure 12: Proposed approached

5.10 Implementation

The implementation was done in C++ and an OpenGL user interface, and was tested on an Intel Core2Duo, 2.8 GHz PC, running Windows 7. The software accepts a polygonal model and creates a hybrid slice model that could be used successfully for CAPP for multi-axis machining process. Figure 13 shows a polygonal model from which different standard and hybrid slice models are created and the corresponding area sliced (considered) for CAPP about three primary axes is shown in table 2. It can be seen in table 2 that for the hybrid slice model all the triangles were sliced as compared to standard slice model. Figure 14 shows different polygonal models for which hybrid slice models and standard slice models were considered and their corresponding sliced area is shown in table 3. This shows that for every polygonal model using the hybrid slice

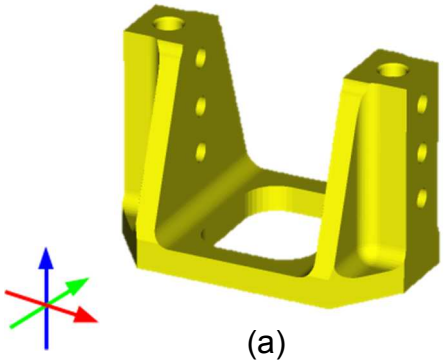
model; it captures the complete surface area and allows for well-informed process planning for multi-axis machining processes.

Table 2: Implementation results (Model in Figure 13)

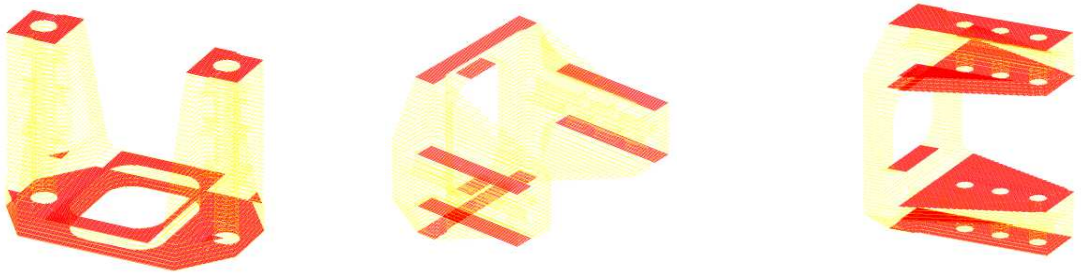
Axis	Total area (in^2)	Standard model Sliceable area (in^2) (%)	Hybrid slice model sliceable area (in^2)(%)
Z	17.8	14.3(80.33)	17.8 (100.0)
Y		15.8(88.76)	17.8(100.0)
X		12.5(70.22)	17.8(100.0)

Table 3: Standard slice model vs Hybrid slice model area

Models (Figure 14)	Dimensions (inch)	Polygonal model	Standard slice model	Hybrid slice model
		Surface area (in^2)	Surface area Sliced (%)	
A	2.67x2.32x0.69	19.38	8.73(45.0)	19.38(100.0)
B	2.3x1.6x1.6	17.82	14.34(80.4)	17.82(100.0)
C	2.5x5.59x0.11	10.27	2.44(23.7)	10.27(100.0)
D	7.4x9.2x14.6	465.13	375.76(80.7)	465.13(100.0)
E	8.9x8.2x9.9	291.25	200.76(68.9)	291.25(100.0)



(a)



Z-axis Y-axis X-axis

(b)

Z-axis Y-axis X-axis

(c)

Figure 13: (a) Polygonal model (b) Hybrid slice model (c) Standard slice model

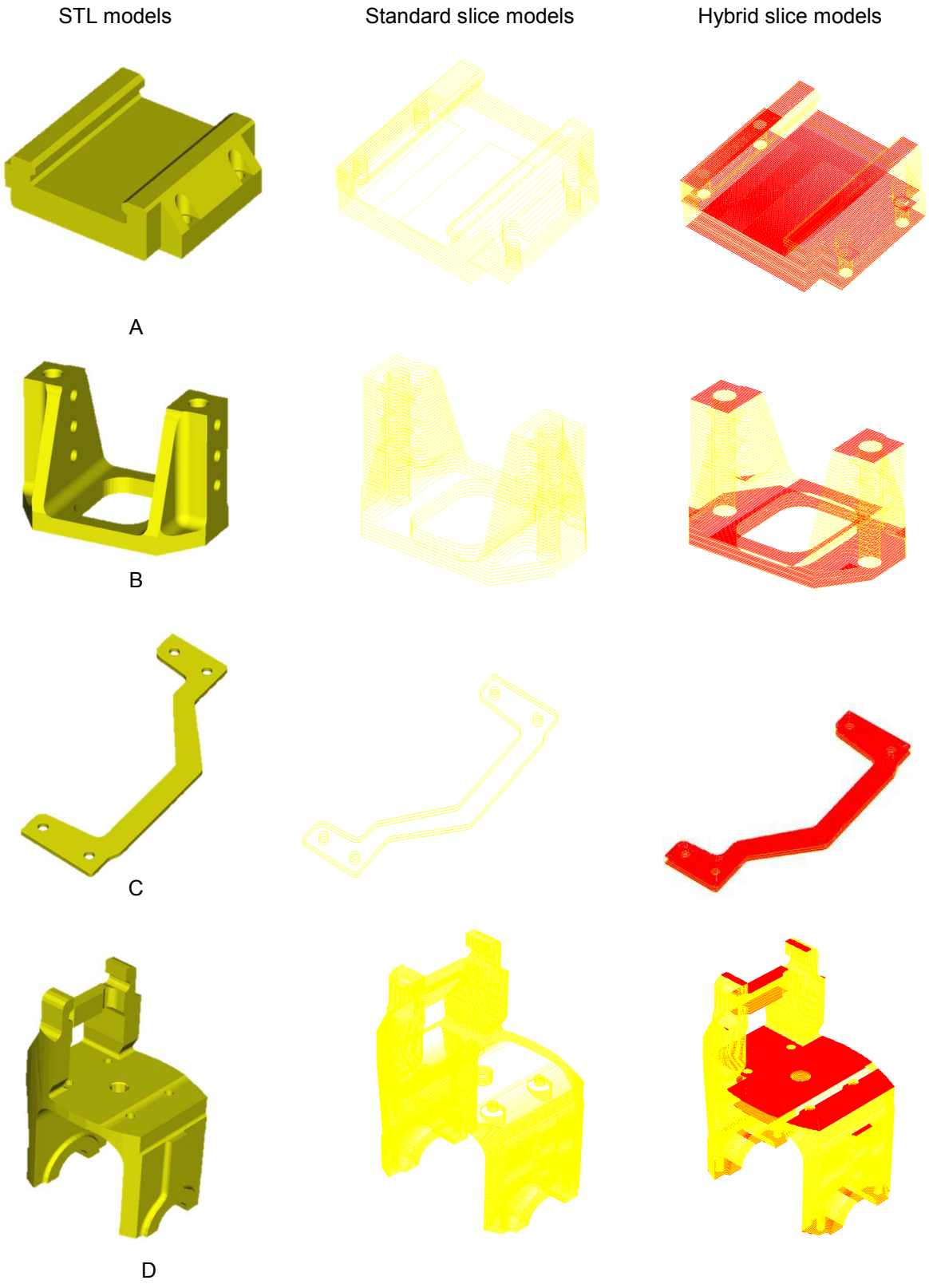


Figure 14: Implementation

5.11 Setup Planning Using Hybrid Model

For machining parts using subtractive manufacturing specifically rapid machining process, Frank et al [19] developed CAPP algorithms for determining line of sight based visibility for each segment present on the slice model generated from polygonal models. The visibility from the slice model would determine machining setups that would create a geometrically accurate part. However these algorithms were restricted to a specific rotary axis for making setup decisions. Chapter 4 hence presented new multi-axes CAPP methods that could provide better solutions for machining polygonal models. In chapter a facet based visibility method was presented where the visibility of each facet on the model could be determined. The visibility of segments that are part of the standard slice model in the hybrid slice model can be determined by using algorithms developed by Frank et al [19] as they are. This chapter specifically proposes algorithms that determine the visibility of newly added parallel segments from the aforementioned hybrid slice model. Once the visibility from standard slice model is

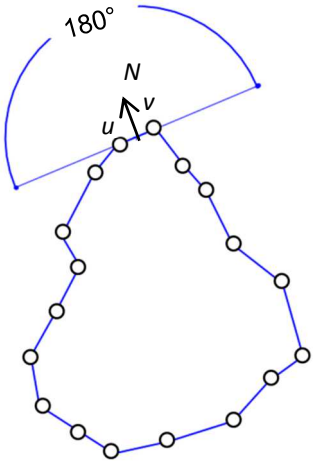


Figure 15: Visibility range for a segment

$$(V_u \cap V_v) = [RV_u, LV_u] \cap [RV_v, LV_v] \rightarrow [RV_v, LV_u]$$

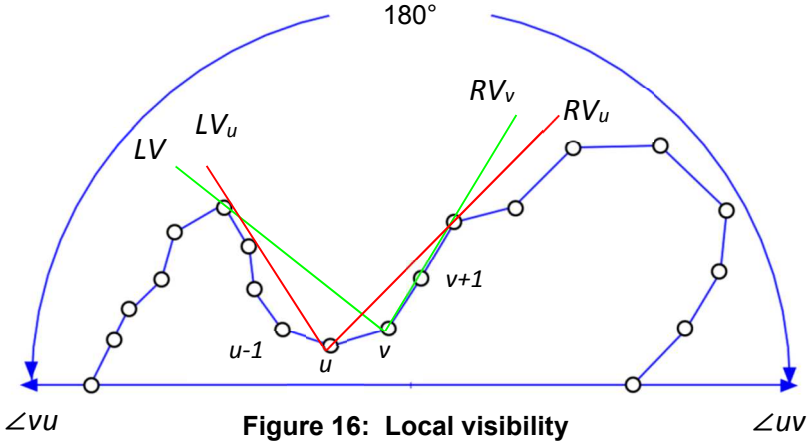


Figure 16: Local visibility

determined, the parallel segments can be combined to give the complete visibility of the hybrid slice model.

5.12 Visibility Of Parallel Segments

Once the visibility is determined, the machining orientations can be chosen using a set cover solution to rapid machine accurate part geometries. In the algorithms developed by Frank et al [19] the visibility of each segment on the standard slice model was categorized into 2 types, a) local visibility i.e. the segment visibility with respect to the chain it belongs to on the same slice (Figure 2) and b) global visibility i.e. the segment visibility with respect to all the chains excluding the chain it belongs to on the same slice (Figure 3). In that the maximum visibility range for any given segment is 180° on its normal (N) side (Figure 15). For example in the local visibility, the visibility range for the segment *uv* with respect to its own chain shown in figure 16 is given by intersecting the right and left visibility ranges of points *u* i.e $[RV_u, LV_u]$ and those of *v* i.e $[RV_v, LV_v]$ on the segment. From figure 16 it can be easily seen that the visibility of segment *uv* would always have bounds RV_v and LV_u that would lie in the maximum range of 180° (Figure

$$(VB_u \; U \; VB_v) = [RB_u, LB_u] \; U \; [RB_v, LB_v] \xrightarrow[N]{} [RB_u, LB_v]$$

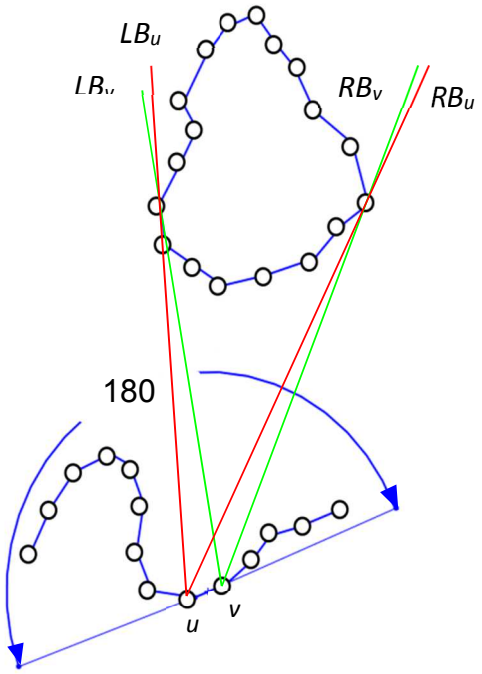


Figure 17: Global visibility

16). Additionally the global visibility of a segment on the slice model is the one with respect to all the chains on the same slice except the chain on which the segment lies. In this category the visibility of the segment on a given chain on a given slice is found out by determining the visibility ranges for a segment blocked by other blocker chains present on the same slice (Figure 17). These visibility blocking ranges are characterized as $[RB_u, LB_u]$ for point u and $[RB_v, LB_v]$ for point v on the segment uv.

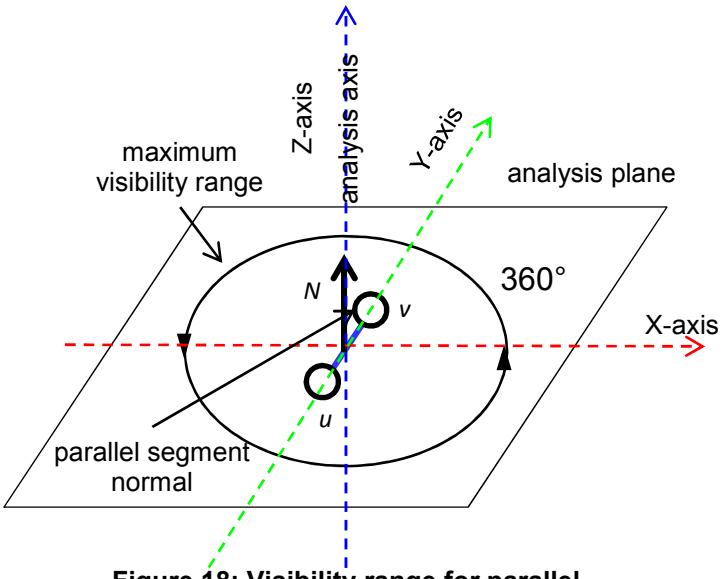


Figure 18: Visibility range for parallel

In addition to the standard slices, the hybrid slice model also contains the parallel segments. In order to perform a correct setup planning using hybrid slice model it is necessary that the visibility of all the parallel segments is also determined. The parallel segments due to their geometric nature lie in a plane parallel to the primary plane using which standard slice model was created. This provides a maximum visibility range for parallel segments of 360° about a true circle perpendicular to the axis about which the standard slice model is generated (Figure 18). All parallel segments remain as individual segments and don't form closed polygon chains like those occurring in the standard slice model. Additionally, being in the same plane the visibility of a given parallel segment with respect to other parallel segments is not affected. Hence when it comes to local visibility the parallel segments have full visibility in the range of 360°.

However at the same Z-height where the parallel segments exist along the axis of standard slices, there could always be other blocker chains from the standard slices that could affect their visibility (Figure 19). Hence the visibility of all parallel segments w.r.t. other blocker chains around them needs to be determined by the global visibility technique. This requires computing the visibility of a segment in a plane about a circular basis w.r.t blocker chains present at the same height (Figure 5). Geometrically, a given parallel segment could have three different types of locations with respect to blocker

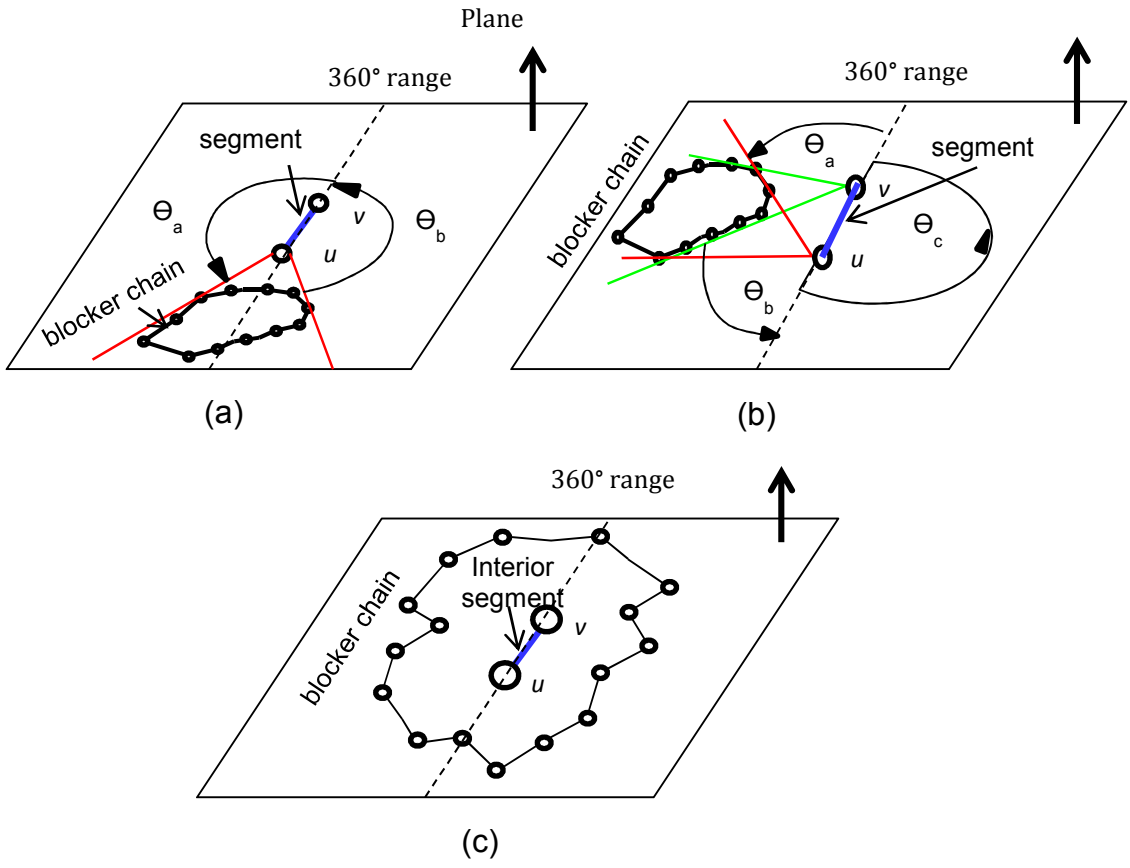


Figure 19: (a) Blocker chain intersecting a parallel, (b) Blocker chain on a side of parallel segment, (c) Blocker chain enclosing a parallel segment

chains at same height, a) a blocker chain present on either or both sides of the parallel segment (Figure 19.a), b) a blocker chain intersecting an infinite extension of the

parallel segment on either or both sides (Figure 19.b) and c) a blocker chain enclosing the parallel segment such that the segment is interior (Figure 5.c). Using the new modified algorithms, the visibility of all parallel segments in the hybrid slice model can be determined. There visibility, combined with the standard slices, provides complete visibility of the hybrid slice model representing the entire model surface for setup planning.

5.13 Implementation

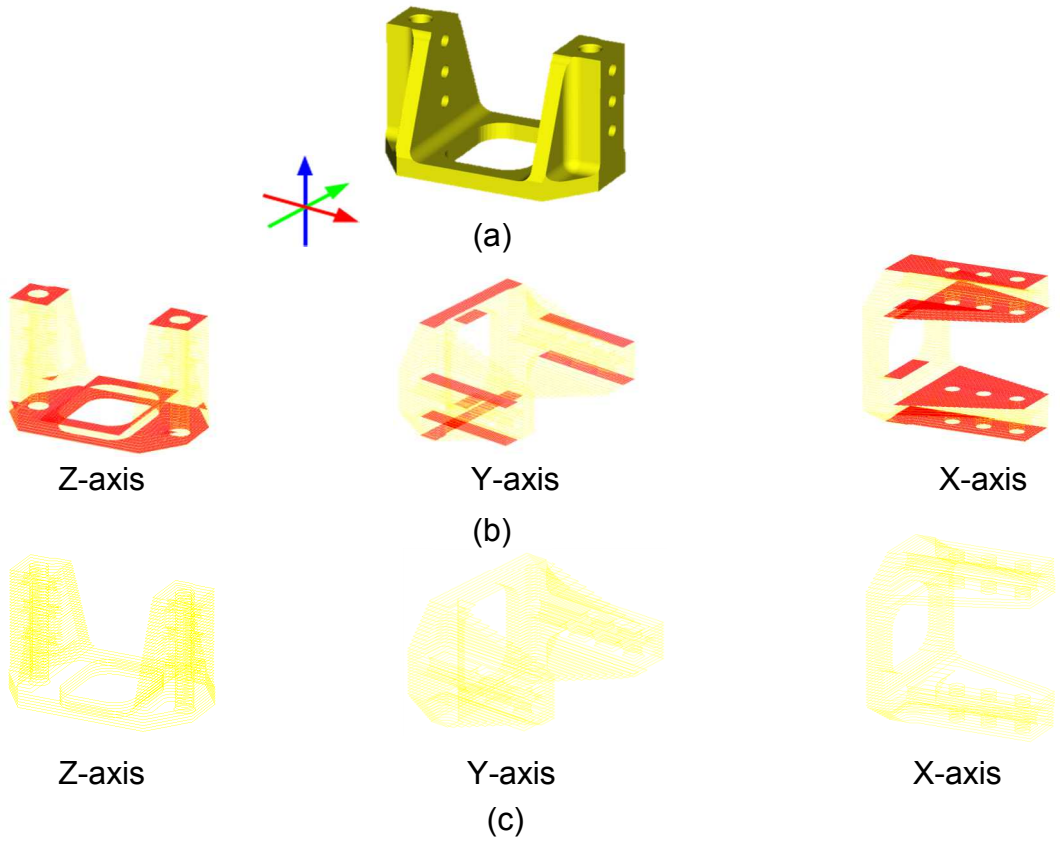


Figure 20: (a) Polygonal model (b) Hybrid slice model (c) Standard slice model

The implementation was done in C++ and an OpenGL user interface, and was tested on an Intel Core2Duo, 2.8 GHz PC, running Windows 7. The software accepts a polygonal model and creates a hybrid slice model that is used for determining visibility of segments present in the hybrid slice model. The visibility from parallel and non-parallel

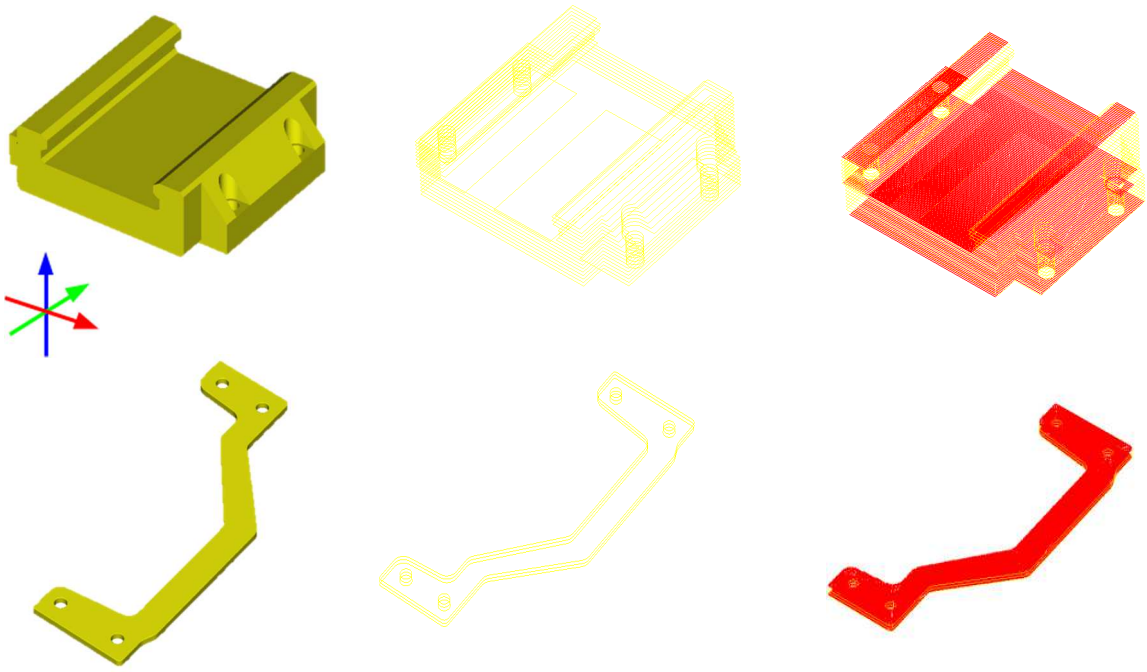
segments is mapped back to the facets present on the polygonal model. This provides the visibility information of every facet present on the polygonal model and in turn the visibility map of the polygonal model. Figure 6 shows a polygonal model from which a standard and a hybrid slice model was created and visibility determined through them were mapped back to the polygonal model. Table 1 shows the results for a polygonal model whose visibility map was determined about three primary axes through both the standard and hybrid slice model. The results show the true visibility map determined due to the use of hybrid slice model, where the complete polygonal model area was considered as compared to the use of standard slice model where partial area is considered. Figure 7 shows multiple polygonal models and their corresponding hybrid slice models where visibility was determined and mapped back to their corresponding polygonal models (Table 2).

Table 4: Standard slice model Vs Hybrid slice model sliceable area

Axis	Total area (in^2)	Standard model		Hybrid slice mode	
		Analyzed area (in^2) (%)	% Visibility	Analyzed area (in^2)(%)	% Visibility
Z	17.8	14.3(80.33)	61.3	17.8 (100.0)	92.4
Y	17.8	15.8(88.76)	81.9	17.8(100.0)	89.6
X	17.8	12.5(70.22)	69.9	17.8(100.0)	87.9

Table 5: Standard vs Hybrid slice model visibility mapping results

Models (Figure 7)	Dimensions (inch)	Polygonal model	Standard slice model		Hybrid slice model	
		Surface area (in^2)	Considered area (%)	% Visibility	Considered area (%)	% Visibility
A	2.67x2.32x0.69	19.38	8.73(45.0)	36.1	19.38(100.0)	97.9
B	2.5x5.59x0.11	10.27	2.44(23.7)	19.4	10.27(100.0)	99.1



(a (b (c

Figure 21: (a) Polygonal model (b) Standard slice model (c) Hybrid slice model

5.14 Conclusion

The methods introduced in this paper ensures capturing the complete surface information of the part for process planning by using hybrid slices created from polygonal models. This allows a reliable use of slice models for CAPP. This improvement could be significant, allowing multi-axes CAPP for manufacturing polygonal models. The visibility algorithms presented allow for determination of facet based visibility of polygonal models through the use of hybrid slice models. The visibility determined by analyzing hybrid slice models can be mapped to the polygonal model. This allows determining the polygonal model visibility about any given axis and hence would enable performing multi-axes setup planning for CNC machining process. Being able to perform CAPP for multiple axes machining process will provide efficient and robust planning of machined parts with higher quality and lower initial cost.

5.15 References

[1] Leong, Mr KF, C. K. Chua, and Y. M. Ng. "A study of Stereolithography file errors and repair. Part-1. Generic solution." *The International Journal of Advanced Manufacturing Technology* 12.6 (1996): 407-414.

[2] Leong, Mr KF, C. K. Chua, and Y. M. Ng. "A study of stereolithography file errors and repair. Part 2. Special cases." *The International Journal of Advanced Manufacturing Technology* 12.6 (1996): 415-422.

[3] Szilvśi-Nagy, M., and G. Y. Matyasi. "Analysis of STL files." *Mathematical and Computer Modelling* 38.7 (2003): 945-960.

[4] Ju, Tao. "Fixing geometric errors on polygonal models: a survey." *Journal of Computer Science and Technology* 24.1 (2009): 19-29.

[5] Borodin, Pavel, Marcin Novotni, and Reinhard Klein. "Progressive Gap Closing for Mesh Repairing." *Advances in Modelling, Animation and Rendering*. Springer London, 2002. 201-213.

[6] Ju, Tao. "Robust repair of polygonal models." *ACM Transactions on Graphics (TOG)* 23.3 (2004): 888-895.

[7] Attene, Marco, and Bianca Falcidieno. "ReMESH: An interactive environment to edit and repair triangle meshes." *Shape Modeling and Applications, 2006. SMI 2006. IEEE International Conference on*. IEEE, 2006.

[8] Bischoff, Stephan, Darko Pavic, and Leif Kobbelt. "Automatic restoration of polygon models." *ACM Transactions on Graphics (TOG)* 24.4 (2005): 1332-1352.

[9] Attene, Marco. "A lightweight approach to repairing digitized polygon meshes."*The Visual Computer* 26.11 (2010): 1393-1406.

[10] Huang, S-H., L-C. Zhang and M. Han. "An effective error-tolerance slicing algorithm for STL files." *The International Journal of Advanced Manufacturing Technology* 20.5 (2002): 363-367.

[11] Choi, S. H., and K. T. Kwok. "A tolerant slicing algorithm for layered manufacturing." *Rapid Prototyping Journal* 8.3 (2002): 161-179.

[12] Sabourin, Emmanuel, Scott A. Houser, and Jan Helge Bøhn. "Adaptive slicing using stepwise uniform refinement." *Rapid Prototyping Journal* 2.4 (1996): 20-26.

[13] Pandey, P. M., N. Venkata Reddy, and S. G. Dhande. "Real time adaptive slicing for fused deposition modelling." *International Journal of Machine Tools and Manufacture* 43.1 (2003): 61-71.

[14] Ma, Weiyin, and Peiren He. "An adaptive slicing and selective hatching strategy for layered manufacturing." *Journal of Materials Processing Technology* 89 (1999): 191-197.

[15] Cormier, Denis, Kittinan Unnanon, and Ezat Sanii. "Specifying non-uniform cusp heights as a potential aid for adaptive slicing." *Rapid Prototyping Journal* 6.3 (2000): 204-212.

[16] Mani, Ka, Prashant Kulkarni, and Debasish Dutta. "Region-based adaptive slicing." *Computer-Aided Design* 31.5 (1999): 317-333.

[17] Chen, X., et al. "Direct slicing from PowerSHAPE models for rapid prototyping." *The International Journal of Advanced Manufacturing Technology* 17.7 (2001): 543-547..

[18] Luo, Xiaoming, and Matthew C. Frank. "A layer thickness algorithm for additive/subtractive rapid pattern manufacturing." *Rapid Prototyping Journal*16.2 (2010): 100-115.

[19] Frank, Matthew C., Richard A. Wysk, and Sanjay B. Joshi. "Determining setup orientations from the visibility of slice geometry for rapid computer numerically controlled machining." *Journal of manufacturing science and engineering* 128.1 (2006): 228-238.

CHAPTER 6. CONCLUSION AND FUTURE WORK

This dissertation presented three new methods for advanced CAPP for multi-axis CNC machining using feature free polygonal models. Chapter three provided a novel method where new algorithms were designed for the setup planning of finish machining for Multi-Surface Parts about a rotary axis. In this setup planning challenge multi-colored polygonal CAD models were used for CAPP where different regions on the CAD geometry were colored different. These colors could represent volumetric or surface attributes such as density variation, material combinations, ductility, and malleability, material coating, texture, hardness, toughness, or roughness, to name a few. These multiple colored regions were used for setup planning such that a custom bone implant could be machined with distinguished sub-surface characteristics, specifically roughness. This in addition to the accurately machined implant geometry could enhance its bio-mechanical stability and provide better trauma treatment in the field of orthopedic medicine.

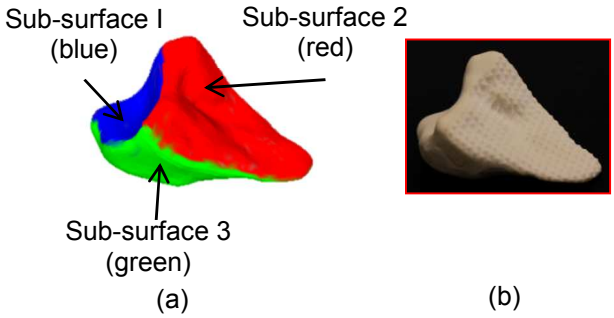


Figure 1: Freeform components (bone implants) (a) Multi-surface implant (b) Machined implant

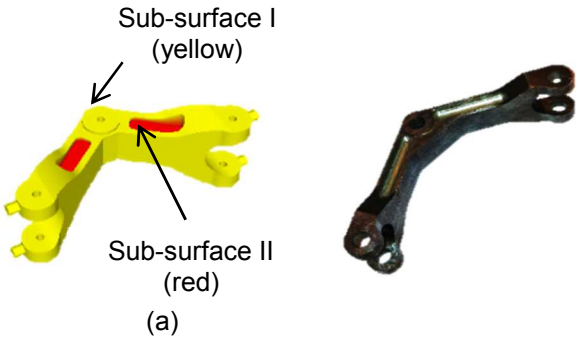
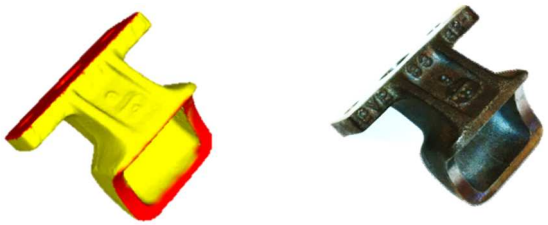


Figure 2: Prismatic components (industrial parts) (a) Multi-surface part (b) Machined part

Additionally these algorithms, with application-specific adaptations could also be used in setup planning for machining multi-surface industrial components. These components could be ones with selective surfaces designated with critical tolerances. In this case, the setup planning efforts would be able to determine setups that could allow creation of these critical surfaces individually.



Colored CAD Machined Casting

Figure 3: (a) Multi-surface CAD part (b) Machined casting

Setup planning using multi-colored CAD models could also be extended to the machining of castings where selective surfaces are to be machined within specific tolerances while leaving the rest of the surfaces as-cast. In this case, the challenge would be to determine setups for machining color designated sub-surfaces that represent critical tolerances while leaving other surfaces un-machined.

Figure 4: Designer feedback

This method could also be deployed for the finish machining of Additively Manufactured (AM) near net-shape components which require some level of post process machining. One proof-of-concept study with a collaborating lab was in the use of post-processing for Electron Beam Melting (EBM) AM components. EBM was used to create a near net-shape component in Titanium, and then rapid machining was used to finish machine several but not all surfaces in order to hold critical tolerances.

Chapter 4 presented a new setup planning approach for the finish machining of free form and prismatic parts using generic feature free polygonal models. These setup planning algorithms are primarily focused on discrete 3-axis machining configurations. Additionally, the objective of the setup planning approach in this method was to consider tool accessibility and line of sight visibility such that the tools chosen in the setup solutions would allow maximum part accessibility and allow machining of the part with higher geometric accuracy. This method mainly considered 3 primary axes for setup planning, with assumptions that prismatic parts are generally accessible about these axes. However any number of off-axes could be considered for setup planning using these methods. This could be particularly

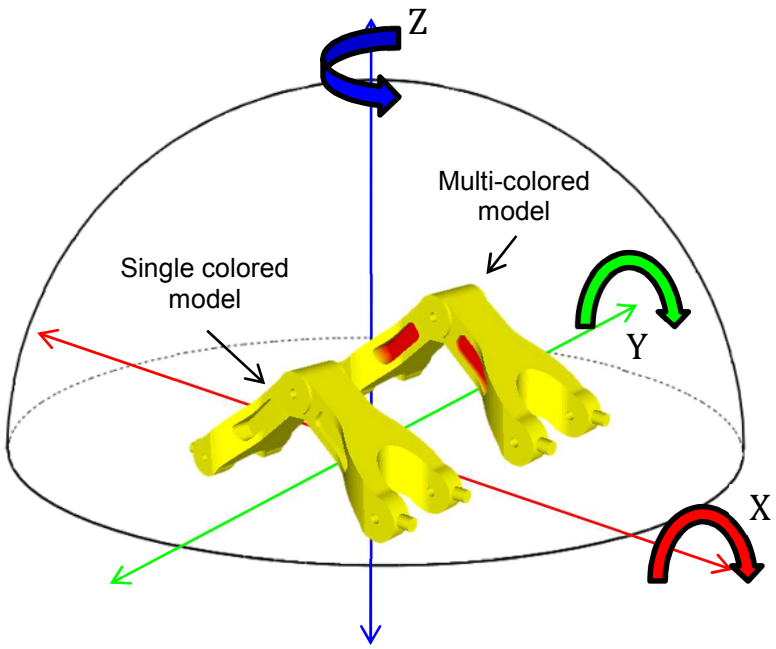


Figure 4: Multi-axis CAPP

beneficial when setup planning is to be done for a freeform object model using multi-axis setups.

Additionally, the algorithms presented in chapter 4 could also be used in an automated manufacturability analysis system. The part model could be analyzed about multiple axes and the analysis results could be provided in the form of user feedback to the designer (Figure 4), enabling a more cost-effective design.

Chapter 5 presented a new method for performing multi-axis CAPP on polygonal models through the use of hybrid slice models. The algorithms presented in chapter three and four benefitted from the use of hybrid slice models. Hybrid slice models and polygonal models could be generated using any surface based CAD format and used for CAPP. The use of hybrid slice models would allow consideration of complete surface data provided on the CAD geometries and enable a complete process plan for multi-axis machining.

Overall, the algorithms and methods presented in this dissertation initiate a step towards advanced CAPP using feature free polygonal models for multi-axis CNC machining processes. These methods could potentially benefit industry by enabling easier process planning for the machining of parts in a geometrically accurate and cost effective manner. Thus the research presented in this dissertation describes an initial step towards the advanced CAPP for multi-axis CNC machining will hopefully aid in developing the next generation of CAPP tools, along with new methods for handling feature-free modeling of component designs.