MODELING HAPTICS FOR VIRTUAL CUTTING EMBEDDED IN CAD

By

HRISHIKESH KATE

A thesis submitted in partial fulfillment of

the requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

WASHINGTON STATE UNIVERSITY

School of Mechanical and Materials Engineering

December 2007

To the Faculty of Washington State University:

The Members of the committee appointed to examine the thesis of Hrishikesh Kate find it satisfactory and recommend that it be accepted.

Chair

ACKNOWLEDGEMENTS

I take this opportunity to thank and acknowledge all those who directly or indirectly helped me and supported me throughout my research.

First and foremost, I am deeply grateful to my advisors Dr. Sankar Jayaram and Dr. Uma Jayaram for their constant attention, helpful guidance and above all undying love and support and for giving me a chance to be a part of the VRCIM lab. I would also like to thank Dr. Lloyd Smith for serving on my committee and providing valuable information and advice whenever needed. I would like to thank all the people in VRCIM lab for their love and support.

I also would like to thank Bob Lentz, Norman Martel from the student's workshop, and Robert "Kurt" Hutchinson and Gary Held for their valuable help during the research. I would like to thank the School of Mechanical and Material Engineering and the graduate school for this opportunity to be a part of this prestigious university which helped me grow overall as a student.

I would like to thank my family and my friends in India for their undivided attention and love throughout my life. I would specially like to thank Neha Waikar, Raghuram Bhaskaran, Okjoon Kim, Nandita Biswas, Constance Bell, Harsimranjeet Singh and Li Juan for their valuable time to help me in my experiments during the research. I won't fail to mention and thank Hrishikesh Joshi & Ritesh Kale for his invaluable help throughout my Masters. Finally thanks to my roommate Rajat and to all my friends in Pullman and Moscow for their love and affection and also the WSU - Pullman Cricket team for making my last three years in the US most memorable.

Last but not the least I would like to thank God for his blessings and giving me all what I have wished for.

MODELING HAPTICS FOR VIRTUAL CUTTING EMBEDDED IN CAD

ABSTRACT

By Hrishikesh Kate, M.S.

Washington State University

December 2007

Chair: Dr. Sankar Jayaram

There has been an ongoing attempt by researchers in bringing virtual models to life with better visualization techniques and numerous force feedback devices. There have also been various attempts at integrating Virtual Reality (VR) with commercial CAD systems.

This thesis presents research related to cutting forces and experiments performed to model these forces to be displayed in virtual cutting integrated inside a commercial CAD system. The research was divided into two phases; first, modeling and visualization of the virtual cutting process within a CAD system and second, implementing force feedback for cutting simulation. The force feedback device used for this research was the PHANTOM® Omni™ from SensAble Technologies.

A virtual cutting capability was implemented inside CATIA V5. This capability allows a user to interactively cut an assembly in the CAD system and perform the geometry manipulations required to cut and separate the parts for manipulation using VR devices.

Experiments were also conducted in an attempt to understand the relationships between cutting velocity, resisting force, material hardness and thickness of the material being cut. The results from the experiments were used to suggest relationships that could be used to provide force feedback for cutting forces using force feedback (haptic) devices. A simple proof of concept was implemented using the PHANTOM® Omni™.

TABLE OF CONTENTS

Appendix C – Experimental Results from Pine samples of Different Thicknesses 102

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

VR - Virtual reality (VR) is a technology which allows a user to interact with a computer simulated environment, be it a real or imagined one.

Haptics – According to www.wikipedia.com, Haptics is the study of touching behavior. ***Haptic*** originates from the Greek word *haptesthai* meaning "contact" or "touch". Haptic technology refers to technology which interfaces the user via the sense of touch by applying forces, vibrations and/or motions to the user.

API (Application Programming Interface) - An API is a source code interface that a computer application, operating system or library provides to support requests for services to be made of it by a computer program.

CAD (Computer-aided design) – CAD is the use of a wide range of computer-based tools that assist engineers, architects and other design professionals in their design activities.

CHAPTER ONE

INTRODUCTION

"Virtual Reality (VR): A computer system used to create an artificial world in which the user has the impression of being in that world and with the ability to navigate through the world and manipulate objects in the world" [1]. There have been ongoing attempts by researchers in bringing virtual models to life with better visualization and interaction techniques and numerous force feedback devices. With new ideas, incorporated with advanced equipment, many have successfully implemented various fascinating applications. VR has evolved through many years of research in hardware and software technologies and applications.

It can be said that VR is a good interface between people and information technology and offers better ways of communication of information, better visualization of processes and creative expression of ideas. VR has been successfully used in training (military, medical, hazardous environments, etc), education, design evaluation, visualization, ergonomic studies, simulation of assembly sequences and maintenance tasks, entertainment and gaming and much more.

VR can be classified into various categories: desktop, projected, semi immersive and immersive. The desktop VR environment is one where the 3D environment is displayed on a 3D desktop computer monitor whereas in the projected 3D environment, the environment is projected onto a big screen which enables a single user to explain and demonstrate the concepts to a group of people. CAVE™ is one of the examples of such an environment where multiple screens are used to surround the user with images. The semi-immersive environment includes simulations of advanced flights, ships and vehicles. For example, while simulating a truck cabin the driver's seat is a physical model, the view of the outside world is computer generated. In the last type, which is the immersive 3D environment, a head mounted display (HMD) is used. In this kind of environment the user is completely immersed and feels like he/she is a part of the system and has no visual contact with the physical world. The HMD has two screens for two eyes. These two screens have the same image at an offset to give a perception of depth to the human mind to view objects in 3D.

VR also incorporates other devices such as gloves with sensors for finger tracking (CyberGlove® from Immersion Corporation). In a typical glove there are 22 embedded sensors to track the motion of the palm and fingers. The motion of the hand can be graphically displayed on the computer screen and is used for grasping and manipulation of objects in VR. To get a more realistic feel during the assembly and disassembly process haptic devices can be integrated with the VR system. Some examples of haptic devices used are CyberGrasp® and CyberForce® systems from Immersion Corporation, and PHANTOM® from

SensAble Technologies. The CyberGrasp and CyberForce, if incorporated together, can give the user a very realistic feeling of an object's weight and increase the sense of touch with good force feedback. On the other hand the PHANTOM® is a small device and generates forces on a pen shaped object that allows the user to feel the physical properties of a virtual object.

A good tracking system plays an important part in VR and provides the necessary position and orientation of objects in virtual space. For example the "Flock of Birds®" from Ascension technologies provides the position and orientation of any object using six distinct variables: three translational and three rotational.

Apart from these devices, to get better uninterrupted and continuous data, computers with faster processors and large memories (RAM) are desired. The selection of a computer system and other devices generally depends on the application of VR. In the case of virtual assembly, the CAD systems would be used for initial modeling and assembling the objects. CAD systems are trying to integrate VR with their basic functionalities to make the task of the end user easier. Because of this the user can model the objects and integrate them with VR in the same CAD system that they were modeled in. One of the examples of such a CAD system is CATIA V5 from Dassault systems.

We introduce and carry out our research aimed at implementing a virtual cutting environment inside a CAD system integrated with force feedback. A typical CAD system is designed for modeling objects and visualization. It is not designed to incorporate Virtual Reality applications inside it. The force feedback system gives the end user a realistic feel of the cutting process as he can sense and feel all the forces during a typical cutting process.

Implementing the virtual cutting process inside a CAD system becomes important and advantageous to the end users because now they can run VR application in the same CAD system where they modeled all the parts and assemblies. With the creation of this unique capability, there is no need for a separate virtual environment. The user has access to this enhanced capability right from within the familiar CAD environment.

The research was divided into two phases; first, modeling and visualization of the virtual cutting process within a CAD system and second, implementing force feedback for cutting simulation.

CHAPTER TWO

LITERATURE REVIEW

Several research groups have focused on haptic interfaces and force feedback during cutting operation. This literature review presents some of the work in the field of haptic interfaces for real time cutting operations in virtual reality.

G. Liu et. al. presented a paper [2] which describes a cutting force model in a dental training system with haptic display capability. The forces used for the haptic interface are obtained from theoretical analysis of the cutting force between dental tools and teeth. Other forces have also been considered. A number of parameters are considered to take all the factors affecting the cutting force into account. They used a six–dimensional force sensor in tooth cutting experiments to identify the parameters. Dentists performing these virtual operations confirm the sense of cutting force to be very realistic. The PHANTOM® is used as the haptic force feedback interface.

Similar work was presented by A. Balijepalli and T. Kesavadas in their paper [3] on a haptics based virtual grinding tool. They have described the creation of a force model to provide the force feedback to the haptic device in order to give a realistic force sensation. Here a tool - workpiece contact force model is developed to simulate resultant haptic force feedback. Their hypothesis states that "*the average time taken to achieve the force is less when pattern of scallops on the work piece is followed as opposed to random grinding*". The authors presented a haptics based robotic path planning and training tool [4]. In this paper they concentrated more on the development of a system for the training of machine operators and path planners. They also professed the fact that force feedback gives better flexibility during process planning. A theoretical force model was created to derive the forces at the end effector arm. Parameters such as grinding wheel width, wheel diameter, machine stiffness normal to the part, wheel surface speed, relative work piece velocity, depth of cut and constants accounting for kinematics of contact and static cutting edge were considered to calculate the normal grinding force using Gatlin's derivation. This equation is valid only for the grinding wheel. However full contact was not completely achieved, only length of contact was considered by Gatlin.

Force feedback in virtual cutting has been used at Toyohashi University of Technology by A. Tanaka et. al [5]. In this paper the objects they used to cut are closed surfaces which consist of numerous triangular patches. The cutting primarily uses boolean operations between the object being cut and the cutting surface which depends on the motion of the cutting blade. They have also employed a force feedback device to "display" the force sensation during the cutting operation. The generated force for the force feedback device is determined by the interaction between the surface of the blade and part of the object being cut. Here they assume that the cutting friction increases in proportion to the velocity of the cutting blade. Torque is also considered along with forces of friction to obtain necessary force feedback. The Phantom® was used as the haptic device for force display in this research.

H. Yau and C. Hsu in their paper [6] on development of a dental training system based on point model have proposed a surfel model. The surfel models are point based 3 D models of complex geometries which have less geometric and topological constraints for developing a dental training system. This model gives a realistic cutting simulation with the use of a haptic device. Some of the advantages of using a surfel model are the ability to model different cutting tools, implementing a virtual tooth model with better visual quality and less memory space required. The tools and the dental model are assigned with material properties to simulate different force feedbacks for different materials. They also propose the use of a 2nd-order "biquadratic" digital filter to reduce the noise and get a smoother force feedback. This gives real touch sensation throughout the training. To obtain and guarantee the required 1 kHz update frequency of the haptic device, the force computation was calculated and implemented separately from rest of the application. Their haptic display mainly depends on the algorithm which generates forces that depend on whether a sculpting or non sculpting mode is used. In sculpting mode the tooth object can be deformed and in the non sculpting mode it cannot and the user can only feel or touch the object. The force generated in the sculpting mode is proportional to the force used to cut the volume and the speed of cut. Here their final equation for the force is a combination of material force and viscous force which can be calculated and displayed by the Phantom®.

D. Weiss et. al. [7] have simulated a tissue cutting process with scissors and have used it with soft tissue models which can be used in surgical simulators. A two dimensional deformable mass spring damper is used to calculate the cutting forces. The first is a large mesh that is used to calculate translational forces. The second is smaller, and is used to calculate cutting forces. A force mesh is created so that the origin of the force mesh's local coordinate system is always aligned with the coordinates in space of the exposed node along the cutting axis of mesh. This method is used on haptic scissors which in turn gives force feedback and thus creates a haptic interface.

Research on the simulation of the milling process for a skull bone surgery with haptic and visual feedback has been described by M. Eriksson et. al. of the Dept. of Mechatronics in Sweden [8]. This is used for training of surgeons to perform complicated surgeries such as removal of cancerous tumors. The data is taken from CT-scans. Haptic rendering algorithms are discussed in their study. There is a haptic loop which is updated at 1000 Hz. The haptic rendering process here is divided into 2 categories, collision detection between the drill and the skull bone surface and generation of force feedback to the haptic interface to get the appropriate force feedback. They discuss three algorithms which can be used for detection of collision between the probe and the VR-object. These algorithms depend on the method used for modeling the probe object as a point, a segment or a 3D object. In their case they model it as a 3D object for realistic force and torque feedback. The PHANTOM® Omni™ haptic device was used for their research.

A fracture mechanics approach for rendering cutting is used at McGill University, Canada [9]. The researchers created a computational model to cut a deformable body. The forces of cutting are calculated by the amount of work done by the tool and they use this to create a mathematical model of the force calculation. To assist their research, they make some assumptions: a) the work piece is free from any residual internal stress and external loading, b) the deformation is purely elastic, c) the displacement of the tool is normal to the contact surface, and d) the cutting displacement is fixed to a cutting axis without any rotations. PenCat/ProTM haptic device (Immersion Canada Inc.) was used to display the forces computed as a function of displacement.

M. Agus et. al. [10] proposed a haptic model of a bone cutting burr and talk about the haptic interfaces used to get the force feedback for training purposes. They present a strategy for collecting experimental data and validating a bone-burr haptic contact model developed in a virtual surgical training system for middle ear surgery. Their approach is based on the analysis of data acquired during virtual and real burring sessions and it involves intensive testing of the surgical simulator by expert surgeons and trainees as well as experimental data acquisition in a controlled environment.

Research at the University of Hong Kong [11] has focused on haptic rendering of the milling process. The forces for the haptic interfaces here are calculated based on the relationship of the material removal rate and machining power. This haptic rendering is updated 1000 Hz. Other related work done in this field of modeling cutting are presented by a) O. Brad [12] that refers to Metal cutting Modeling b) W. Lai [13] talks about modeling cutting forces in his research, Modeling of Cutting Forces in End Milling Operations and c) C. Yalcin et. al. [14] proposed real time calibration of cutting for CNC milling.

H. Joshi has successfully implemented a Virtual tool environment inside a CAD system in his research [15]. Features like assembly and disassembly and tool operation interfaced with Virtual Reality inside a CAD system (CATIA V5) have been created.

Almost all the literature discussed so far deal with simulating the cutting operation with models, then retrieving the data from the model to calculate the forces of cutting and using haptic devices to provide force feedback based on these models. The research on which we are focusing is on similar lines, in which we wish to model the forces in such a way that when complex assemblies are cut inside a CAD system, the appropriate cutting force can be obtained and displayed based on the geometry and materials properties of the various parts involved during this simulation. Complex assemblies here refer to assemblies that are complex from a haptics point of view. That is, they will have numerous different components of different materials and different geometries. We wish to simulate a realistic cutting effect for the end user for such events, so that when the end user is cutting through an assembly with different components he should feel the difference in the forces which resist the motion of cutting, which will be different for different materials according to their material and geometric properties.

CHAPTER THREE

PROBLEM DEFINITION AND SCOPE OF WORK

3.1 Introduction

The main aim of this research is to investigate topics related to simulating cutting using hand held tools in virtual reality with force feedback inside a CAD system. To obtain this goal it is very important to understand the dynamics of the cutting tool with respect to the work piece. This involves understanding the various forces of cutting and the factors that affect the forces involved in cutting. This also involves understanding the integration of these methods inside a CAD system. This chapter describes the technical challenges to be addressed and scope of the research.

3.2 Challenges

3.2.1 Cutting forces

There are several different key factors that affect cutting and are very important for simulating the complete cutting process. These factors include: feed speed of the cut or the velocity of the cutting tool in the direction of cutting (linear velocity), number of teeth on the cutter, rpm of the cutter in case of a circular saw and strokes per min in case of a jig saw, material properties of the cutter, material properties of the work piece, horse power of the motor of the saw, material removal rate, sharpness of teeth etc. [16] The linear velocity is important because the time of the cut depends on it. The number of teeth on the cutter also plays an important part in the modeling cutting process as the work piece is being divided, because more teeth means less wear and tear on each tooth and an extended life for the cutter. This also means more work done by the cutter in one revolution or one stroke. Similarly, the rotational speed of the cutter, which is often termed as the rpm (revolutions per minute) governs the speed of cutting. The faster the rpm, the faster the cutting speed. The case is the same with the power of the saw. Two of the most important factors are the material properties of the cutter and the work piece. The time of cutting depends on relative material properties. For example if we have a cutter with a carbide tip which is very strong and we have a wooden block and a iron block of same dimensions, the time taken for cutting the metal piece is more than that for the wooden piece, because iron is harder than wood. All the above factors have to be addressed and considered in modeling forces to be simulated in VR.

One of the other challenges we face is in modeling the cutting of multiple materials at the same time. The work piece on which we have to implement our force equation may consist of different materials ranging from soft foams to relatively much harder material as shown in Figure 1 below.

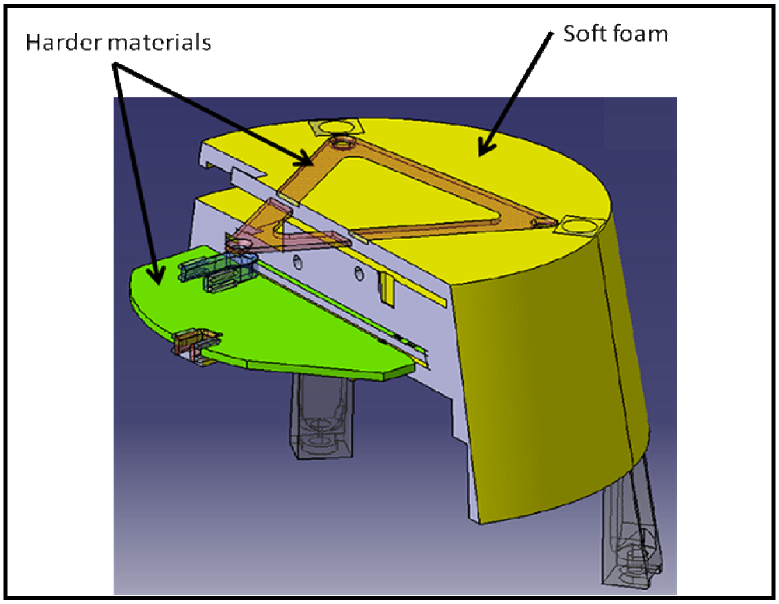


Figure 1 **Cross sectional view of a complex work piece**

There are essentially two situations while cutting. These are,

1. When just one material is being cut.

2. When multiple materials are cut together.

These factors have to be taken into account while coming up with a math model for force feedback, which can be later used and implemented with haptic devices to provide the necessary force feedback.

3.2.2 Implementing inside a CAD system

The next set of challenges, were related to the implementation of the whole process inside a CAD environment. The challenges can be further classified into the following sub categories.

a) Integrating VR with CAD

There are numerous challenges while integrating VR with CAD. The first and foremost among these is that CAD systems are basically designed for geometric modeling of objects and related applications. CAD systems are not designed for real-time interactive graphics integrated with real-time geometry manipulations.

CAD systems basically use mouse and keyboard for user interface, whereas VR system use sensored gloves, trackers etc. Hence the mode of interaction between the end user and the CAD system changes due to the integration of VR. Also the 2D interaction changes to 3D interaction inside VR, as perception of depth is one of the key factors in VR.

The biggest challenge while implementing VR in a CAD system is the ability to extend the capabilities of the CAD system. A typical commercial CAD system has an inbuilt API (Application Programming Interface) system. Most of the CAD systems open some of these API calls to the programmers to modify the application or come up with a new application. It becomes challenging if all the

API calls of the CAD system are not open for external applications. Secondly, these API calls of a typical CAD system are not designed for real-time simulations.

b) Geometric modeling the cut

A typical CAD system is designed for modeling parametric feature based models. When the virtual cutting tool passes through the assembly to be cut, it becomes a challenge to represent the cut inside the CAD system and to use the solid modeling kernel to update and maintain the geometry of the cut pieces.

c) Separation of cut pieces

Another challenge is in separating two cut pieces into two independent pieces, so they can be manipulated separately once the cutting process is over. CAD systems organize assemblies in a tree structure consisting of parts and sub-assemblies. When an assembly is cut using the cutting tool, methods need to be implemented to add the new fragments to this tree. This situation is even more complex when the same component is used as multiple instances in the assembly.

3.3 Hypotheses

3.3.1 Cutting force

Considering all the discussions presented above we put forward a hypothesis that, “During a cutting operation, the linear speed of cutting will increase until the cutting force equals the resistive force”. From the point of view of simulation, this hypothesis, combined with its relationship with all other factors affecting cutting forces, will enable us to derive/propose a force model for haptic feedback.

3.3.2 Integrating with CAD

Considering all the challenges and issues with the final simulation of the virtual cutting process with CAD we propose that “Typical CAD systems will be able to support the geometric modeling needs and the interaction needs of a VR – based cutting simulation”.

3.4 Scope of Work

The scope of work for this thesis is as follows:

3.4.1 Study methods in CAD

Since the virtual cutting procedure has to be implemented inside a CAD system, it is necessary for us to study different methods inside a CAD system for obtaining various data from the models. Some of the topics to be studied are: assigning material properties to the different components included in the assembly, getting the thickness information of each and every component and study different methods to update the collisions between the cutting object and the work piece and obtain data related to the collisions every frame.

3.4.2 Implement a proof of concept using CATIA V5

In this a proof of concept of embedding VR – based cutting simulation inside a CAD system will be created. The feasibility of such implementations will be discussed using this proof of concept.

3.4.3 Use a haptic device to simulate cutting forces

To give the end user a realistic sensation or feel of the cutting process inside VR it is important to display these forces properly. Thus implementing and rendering these simulated forces with help of a haptic device is our next goal. To attain this goal it is necessary for us to study the hardware and software functionality of the haptic device in detail.

3.4.4 Experimentation

For rendering and implementing the cutting forces with the force feedback system it is necessary to model forces. To obtain these forces we need to perform experiments. These experiments will be aimed at finding load (avg. force) vs. avg. velocity relationships for different samples of material with different material properties and different thicknesses. This will help us understand and propose an approximate mechanism for simulation of cutting operations with respect to material properties as well as thickness of the material.

3.4.5 Create of math models to simulate forces

The forces which have to be used for the force feedback device will need to come from a mathematical model based on empirical data. This model will help us calculate the forces for different materials, i.e. if a given work piece has more than one material at different layers then the cutting will be affected by the material properties along with the geometry of each material.

CHAPTER FOUR

INFORMATION FROM CAD MODEL

Getting information from the CAD model interactively, while the virtual model of the work piece is being cut, is one of the important things to be understood and implemented. This chapter talks about methods in detail which were used in an attempt to obtain the information mentioned above and the problems faced while doing so.

The basic idea in getting the information from the CAD model is to obtain the area of contact on the work piece with the cutting tool at any given instance of time. Our first attempt in doing so was by modeling a simple rectangular block using CATIA V5 on which we could perform our initial tests. We also tried similar methods with Pro-Engineer.

4.1 Modeling with CATIA V5

Our initial experiments for getting the information from CAD models were carried out with CATIA V5 where we started by modeling a simple rectangular block. The rectangular block acted as the work piece and we used a datum line as the cutting tool. This is done to obtain the length of the datum line in contact with the block; this distance in turn represents the thickness of the material in contact with the cutting tool.

The thickness was calculated by measuring the distance between the two points, which are essentially the two intersections of the line and the outer surfaces of the block. In simpler terms this would be the distance between the point where the line enters the block and the point where the line leaves the block. Figure 2 below shows these two intersecting points.

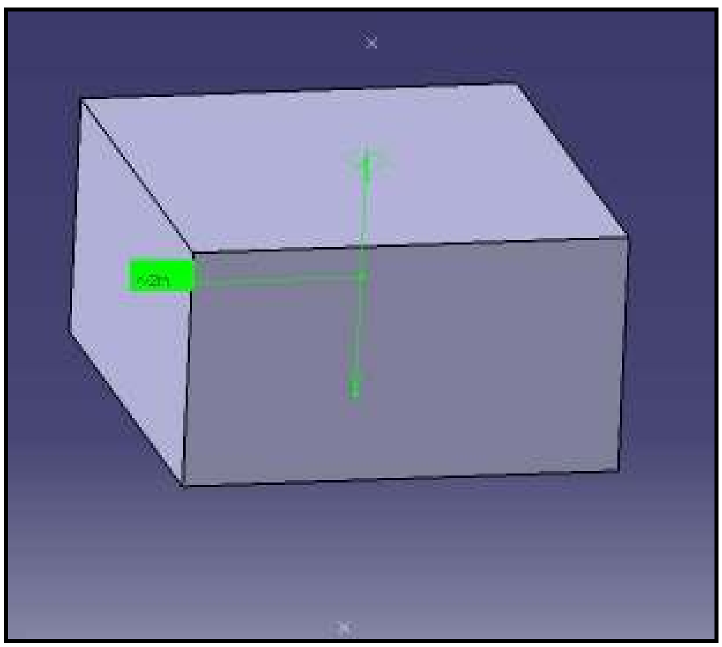


Figure 2 **Simple block in CATIA V5 with the measured dist. between the two intersecting points**

In this case, to draw a datum line in CATIA V5 we have to specify two distinct points on either ends of the block so that the line intersects the rectangular block. Once this is done we need to put two datum points at the intersection of the datum line and the block surface. The problem we faced while doing so was that inside CATIA V5 there are limited options to place a datum point and we did not have the option of placing the datum point on the given intersection, which was our initial idea. We had planned to measure the distance between these two intersecting points and then move the line so that in turn the two intersecting points would be updated and hence a new measurement is obtained however, this did not happen in our case. Therefore, we decided to take the measurement between the two intersections directly and thought of making a macro out of it. In this particular case when we moved the line, the measurement between the two points wouldn't update and we had to take a new measurement every time we moved the line. Figure 3 illustrates such a case.

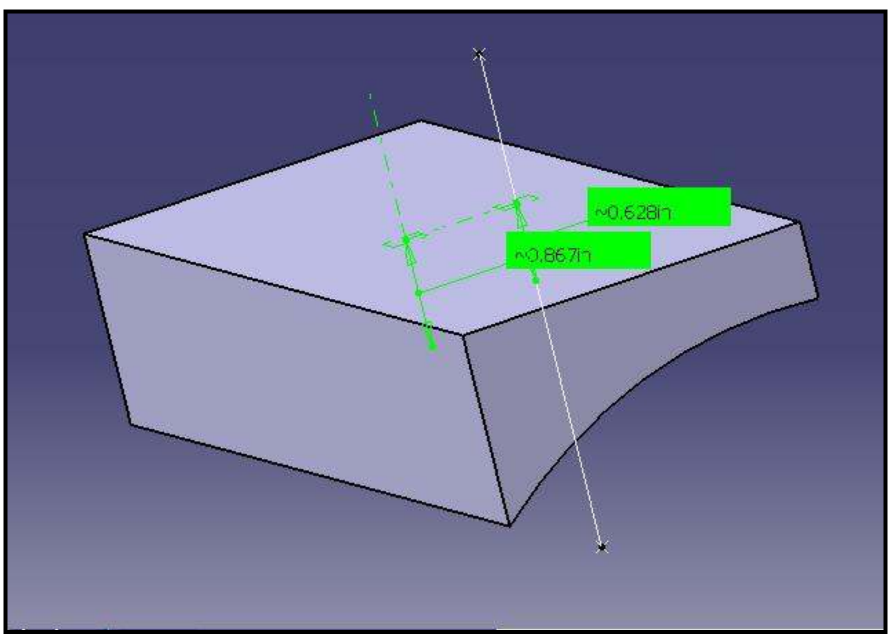


Figure 3 **<u>CATIA V5 does not update the measurement between intersecting points even if the intersecting line is moved to a new position</u>**

We also tried the above procedure on a block with a variable cross-sections area as shown in Figure 3. Here we found out that even if the cross section changes at every instance, the CAD program calculates the exact distance between the two intersecting points.

The next step in this process was to get the measurement at every instance as the cutting tool moves forward to cut further. Hence we had to come up with a procedure so that we can move the datum line along a cutting path and get the measurement between the two intersecting points. This could be then captured as a macro and be used in further code in tandem with the haptics, where the CAD model would give the information to the haptic device about the thickness of the material cut at every frame. We couldn't use this approach because we weren't able record a macro in which the distance between the two intersecting points would update automatically every instance.

4.2 Modeling with Pro/Engineer

Our next approach was to perform a similar kind of procedure using Pro/Engineer; a different CAD system. The base CAD models which would be used as our work piece remained the same from the earlier case. We found out that the advantage of using Pro/Engineer over CATIA V5 is that the distance between the intersecting points updates automatically as we move the cutting tool (in this case just a single line) to a new position and we don't need to measure the distance every time. Figure 4 shows of the model in Pro/Engineer.

This makes our task easier as in this case we don't have to worry about measuring the distance between the intersecting points every frame. The reason we did not use this approach was that we could not record a Macro of the whole process, which was our desired output and hence was discarded.

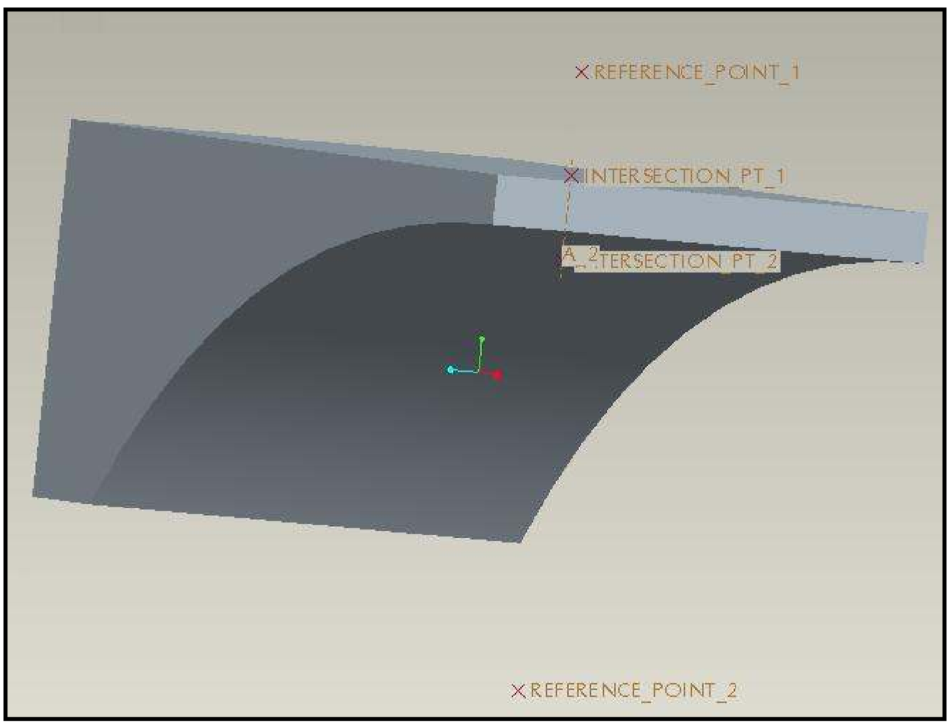


Figure 4 <u>Pro/Engineer model of variable cross section</u>

CHAPTER FIVE

VR CUT – PROOF OF CONCEPT

5.1 Introduction

Our basic goal for the proof of concept is to simulate a fully functional virtual cutting model with cutting tools and assemblies inside a CAD system. The proof of concept put forward takes care of the visualization and geometric modeling parts of the complete virtual cutting process. We divide the whole visualization process into three phases: a) passing the cutting tool through the entire assembly to show the cutting process, b) getting the information from the assembly which was cut and displaying the cut at the locations where the cutting tool and assembly collided during the first phase, and c) separating the two halves of the assembly on either side of the cut to be two completely independent halves which then can be picked separately by the end user.

5.2 VRTools version 1.0

VRTools is a fully functional module designed and implemented *by Integrated Engineering Solutions [17].* VRTools workbench is implemented inside CATIA V5 and it assists the end user with the capability of manipulating different assemblies. This workbench has a set of inbuilt tools which can be easily accessed along with new tools as required by the end user based on the application. This allows a user to assemble and disassemble using tools on fasteners in a VR environment embedded in CAD. We used this workbench for our proof of concept of virtual cutting. The VRTools workbench gave us a platform of virtual environment to show the cutting process.

5.3 Using voxels for cutting the model

An extensive literature review was conducted in order to discover what research had been done in the field of virtual machining. Most virtual cutting simulations are performed using a voxel structure, where the part body is broken up into small individual cubes to provide an approximate representation of the geometry. This simplifies the cutting algorithm because when the cutting tool collides with one of these voxels it is simply turned off. While the voxels do not represent the geometry exactly, they can be reduced in size to give a better volumetric representation. This method is generally chosen because altering the geometry of a B-rep model, used in most CAD systems, is extremely complex and comes at a high computing cost.

To further investigate the use of voxels for cutting simulations, we created a simple test-case voxel cutting program. This utilized functionality from the Virtual Hand SDK toolkit provided by Immersion Corporation as well as basic OpenGL commands inside Microsoft Visual C++. The part to be cut was simply a large rectangular body composed of a variable number of smaller voxel cubes. The program allowed the user to change both the overall dimensions of the rectangular part body as well as the resolution in terms of voxels per side. A conversion program was used to import a CAD model of a circular saw, created in CATIA, into an OpenGL viewing space. The circular saw was given the translation and orientation of a 6 DOF tracker for use as a cutting tool. Knowing the location of the origin of the saw, we came up with an algorithm to check if a given number of points on the curved cutting edge of the saw blade intruded on the volume of each voxel. If the cutting point is found to be within the volume, the voxel is cut and therefore not displayed.

5.4 Using CAD supported methods for cutting the model

CAD systems have significant capabilities in geometric modeling. We wanted to investigate CAD supported features that could be used to cut parts and assemblies with other parts. There were two avenues investigated:

a) Cut with a feature

b) Cut with a part

In order to cut a part with a feature, the swept volume of the cutting surfaces needs to be used to create the "cut" feature. This process was tried with CATIA V5 using a cylinder to represent a saw blade. Unfortunately a complex

3D surface cannot be used in a “sweep” feature which limits the cross – section of the sweep to a rectangle only. This will cause problems when the cut ends in the middle of the part instead of going right through the part.

For cutting a part or assembly with another part, we learned that there were several Boolean operations that could be used. The “Remove” feature simply extracts the intersecting geometry of one part from another. Below, in Figure 5, a wheel is intersecting a block. A remove feature is applied and the resulting geometry shows the intersection of the two parts being deleted, Figure 6.

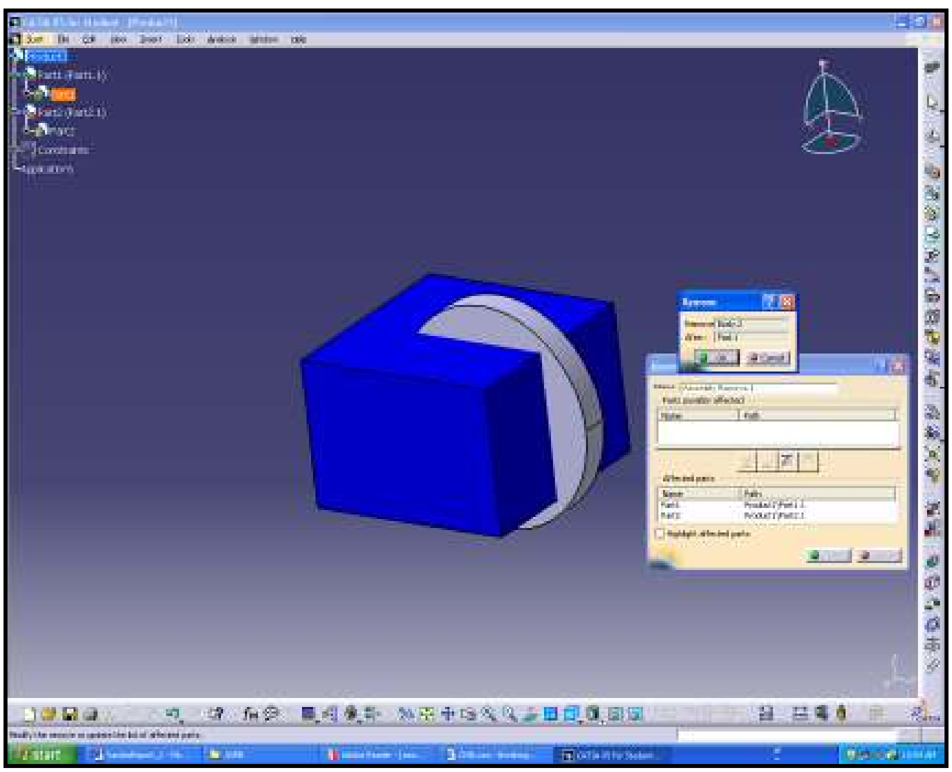


Figure 5 <u>Two intersecting geometries</u>

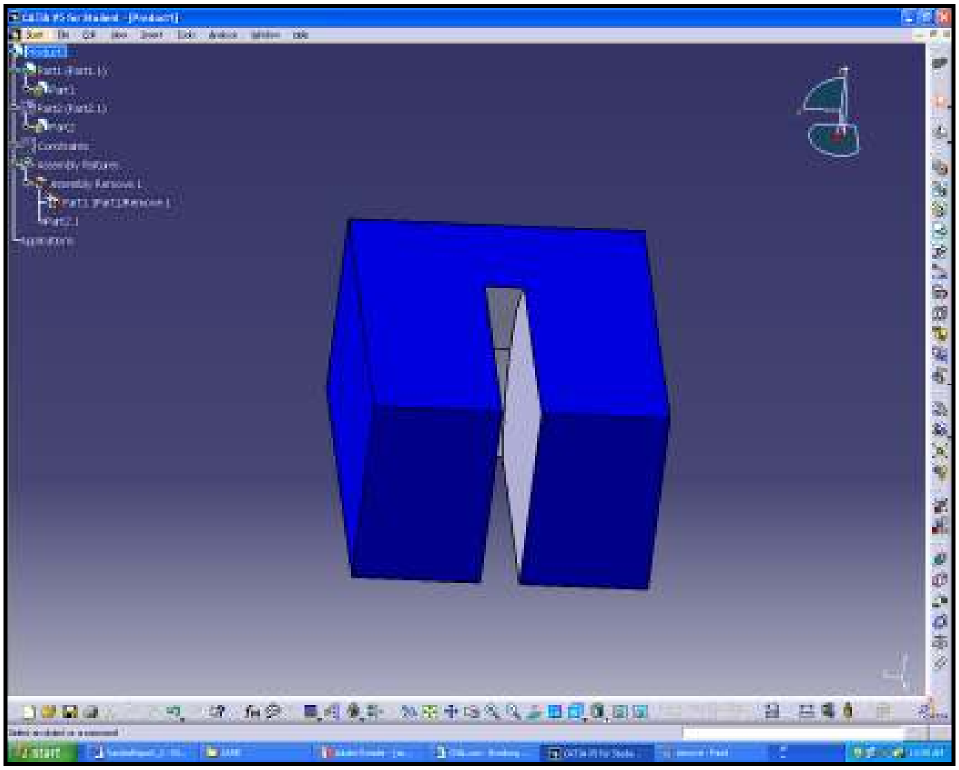


Figure 6 **<u>After applying remove feature to the block</u>**

Using this feature, a macro was created that began with a circular saw outside of a metal plate. Once recording began, the saw was progressively moved into the material to simulate a cutting motion. At the same time, a hidden part consisting of a swept circle was moved in conjunction with the saw. This hidden part had a "Remove" feature with the metal plate, so that as the saw progressed material was removed along with it. After the part had been cut into two pieces, the "Remove Lump" feature was used to separate the half of the part body that was no longer connected. This macro was purely for demonstration purposes and was not interactive.

Once this program was completed, we realized that it is possible to sweep the saw laterally and perform cuts not along the cutting axis. Therefore, we decided to apply a planar constraint so that once the saw entered the work piece its motion would be restricted to a plane parallel to the cutting axis.

5.5 Implementation

After the preliminary research we learned that the remove lump feature discussed above along with macros could be used to get the desired simulation. This feature helped us to keep the simulation simple and clean.

To implement the Boolean operation of remove lump, two instances of the part to be cut were placed in the assembly. These parts were overlapping each other and hence could be seen as one. Next step in the process was recording macros to capture the remove lump feature. Two macros were recorded; one for each instance of the part.

The macro was recorded with the "faces to keep" option. The difference between the two macros was that the faces selected on each part piece were different and on opposite ends. This was done basically so that when you cut the two overlapping foams the two faces lie on either side of the cut.

The two macro codes which were recorded inside CATIA V5 are shown in Figure 7 and Figure 8.

```
Language="VBSCRIPT"

Sub CATMain()

Set partDocument1 = CATIA.ActiveDocument

Set part1 = partDocument1.Part

Set bodies1 = part1.Bodies

Set body1 = bodies1.Item("0_0")

part1.InWorkObject = body1
Set shapeFactory1 = part1.ShapeFactory

Set trim1 = shapeFactory1.AddNewTrim(body1)

Set shapes1 = body1.Shapes

Set remove1 = shapes1.Item("Remove.1")

Set reference1 = part1.CreateReferenceFromBRepName("RSur:(Face:
(Brp:(Pocket.2;0:(Brp:(Sketch.3;4)));None:();Cf11:
());WithTemporaryBody;WithoutBuildError;WithSelectingFeatureSuppor
t;MFBRepVersion_CXR14)", remove1)

trim1.AddFaceToKeep reference1

part1.Update

End Sub
```

Figure 7 Macro 1 for implementing the remove lump feature

```
Language="VBSCRIPT"

Sub CATMain()

Set partDocument1 = CATIA.ActiveDocument

Set part1 = partDocument1.Part

Set bodies1 = part1.Bodies

Set body1 = bodies1.Item("0_0")

part1.InWorkObject = body1

Set shapeFactory1 = part1.ShapeFactory

Set trim1 = shapeFactory1.AddNewTrim(body1)

Set shapes1 = body1.Shapes

Set remove1 = shapes1.Item("Remove.1")

Set reference1 = part1.CreateReferenceFromBRepName("RSur:(Face:(Brp:
(Pocket.5;0:(Brp:(Sketch.6;2)));None:();Cf11:
());WithTemporaryBody;WithoutBuildError;WithSelectingFeatureSupport;
MFBRepVersion_CXR14)", remove1)

trim1.AddFaceToKeep reference1

part1.Update

End Sub
```

Figure 8 <u>**Macro 2 for implementing the remove lump feature**</u>

5.6 Implementation of cutting in CATIA V5 using CAA

When the assembly is in place with the hand object inside the CAD system for the final manipulation, we start our cutting process. This program code for the cutting process needs to be programmed in such a way that it will follow these steps

1. Sweep the cutting tool through the assembly
2. Separate the assembly into two halves
3. Make the two halves of the assembly independent of each other for further manipulation.

- Sweep Process :

The first step is to pass the cutting saw through the assembly to be cut at a realistic speed (about 3 seconds to sweep through the given assembly). During this process the program calculates and stores the different locations on the path which the cutter passed through during the sweep process. This data was captured once every 200 graphics update frames in the form of transformation matrices.

- Separating the assembly into two halves to show the cut:

Once the saw cutter passes through the assembly we select the "start manipulation" button on the VRtools workbench in CATIA V5. The program then starts adding the cutter instances to the assembly at the locations which were recorded during the sweep process. The program also sets the cutter instances to be hidden so that it cannot be seen by the end user to keep the realistic feel. The program now starts checking for collisions between the cutter instances and the assembly. The volume of collision from the assembly is now subtracted from the actual assembly so that the cut is modeled.

- Separating the assembly into two independent halves for further manipulation:

The next step in this process is to pick up the two disjointed halves of the assembly separately. The process of finding the collisions and subtracting the assembly doesn't necessarily cut the part in two independent halves. Hence we need to have two parts overlapping at the same location before starting the sweep with the cutting saw. Even in this case we follow the same process of the sweep and checking collisions as well as subtracting colliding area from the part and we get the same result. But in this case we have two different parts at the same location. The above process divides the parts into two halves, let's assume half 1 and half 2 on both assemblies. In this case the program will delete half 1 from part 1 and half 2 from part 2 before displaying the cut. The end result of this process gives two halves after cutting, but here the two halves are independent from each other as they come from different but identical parts and can be picked up separately. The functionality used in the program to perform this step of separating the part into two separate halves is a function from CATIA V5 called "faces to keep" which in turn is a sub-function of "remove lump" function. Once this is done the two halves of the part can be treated and manipulated independently and hence can be picked up by the virtual hand separately.

Figure 9 shows a flowchart of the complete cutting process.

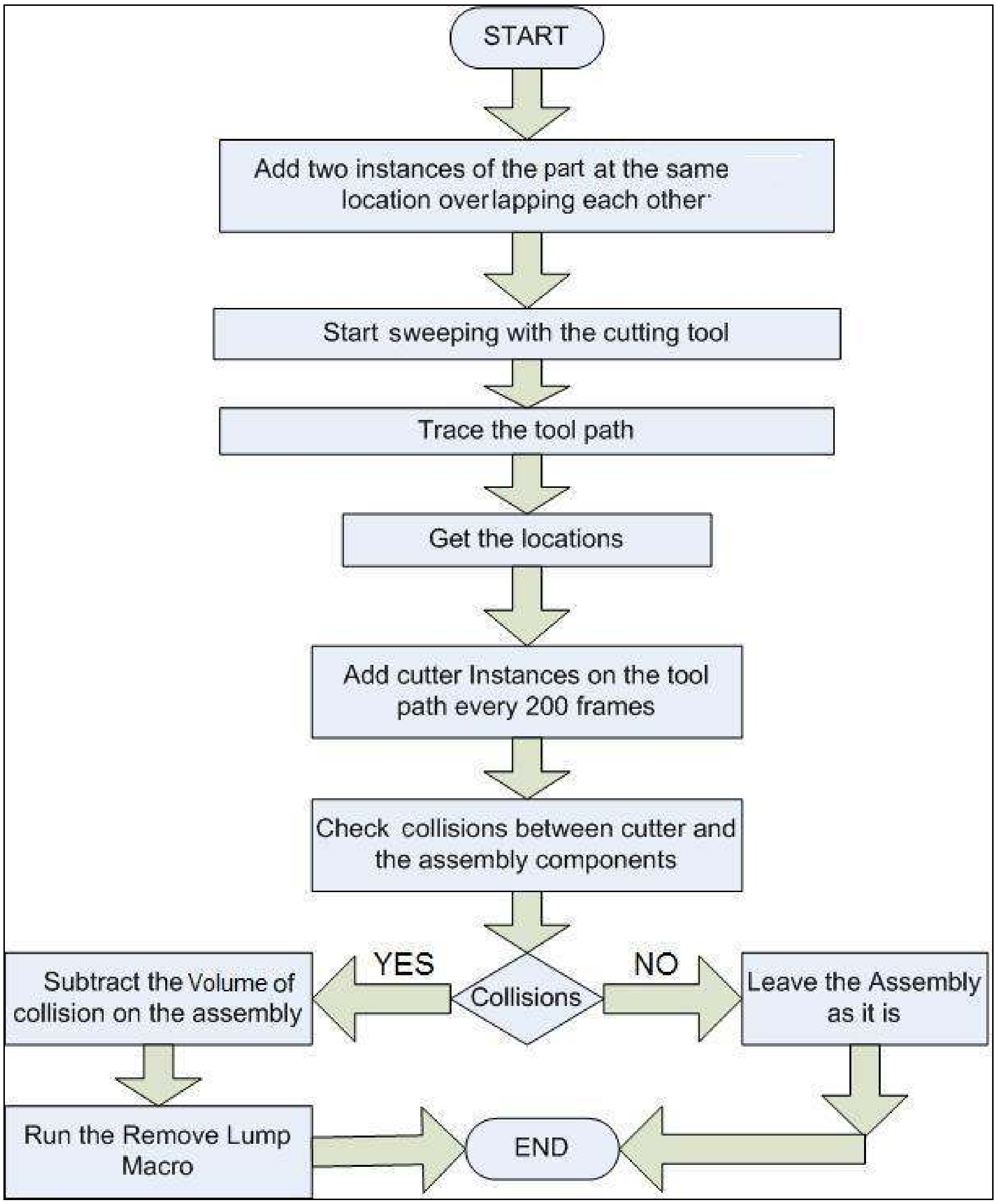


Figure 9 **<u>Flowchart of the cutting process code</u>**

5.7 Using the implementation

Once the macros were created and implemented the procedure for using the surface was simple and straight forward. The procedure included the following steps.

1. Go to the VRTools Workbench.

   Start → Infrastructure → VRTools

2. Click on the Green Arrow on the right side menu last button to start VRTools Manipulation Figure 10.

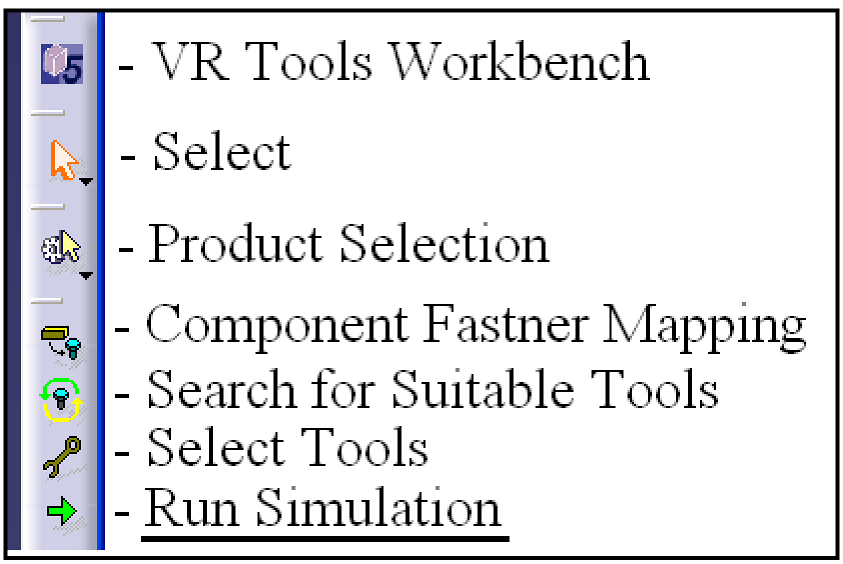


Figure 10 VR Tools simulation menu button

Once you press the "run simulation" you will see the window called "Simulation" as shown in Figure 11. Check on "Ascension Bird" and “Cyber Glove”. Also you can change the size of the hand by changing the “Hand Scale Factor”. Once all this is done, proceed by pressing start simulation.

Figure 11 **<u>VR Tools simulation Window</u>**

<u>Note</u>: Make sure that the button on the glove is “Switched Off” before you Start Simulation

3. Once the start simulation is pressed, grab the cutting tool and make sure that your thumb and two other fingers touch the tool to pick it up properly.

4. Go close to the work piece so that you are just outside the work piece as if you have to begin the cut at that point Figure 12. “Switch On” the button on the glove and start cutting. Optimize the speed to be about 2-3 seconds to go through the workpiece Figure 13.

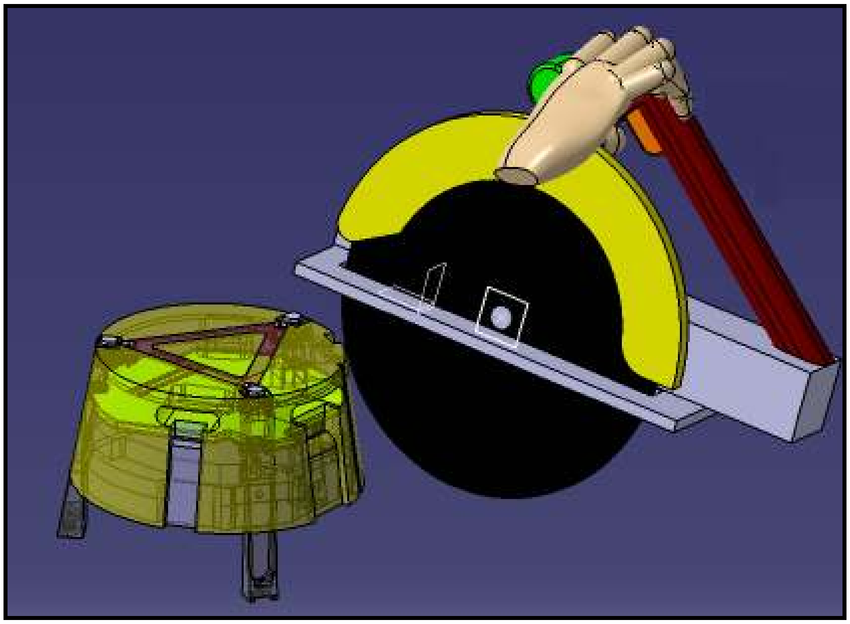


Figure 12 Before cutting the assembly

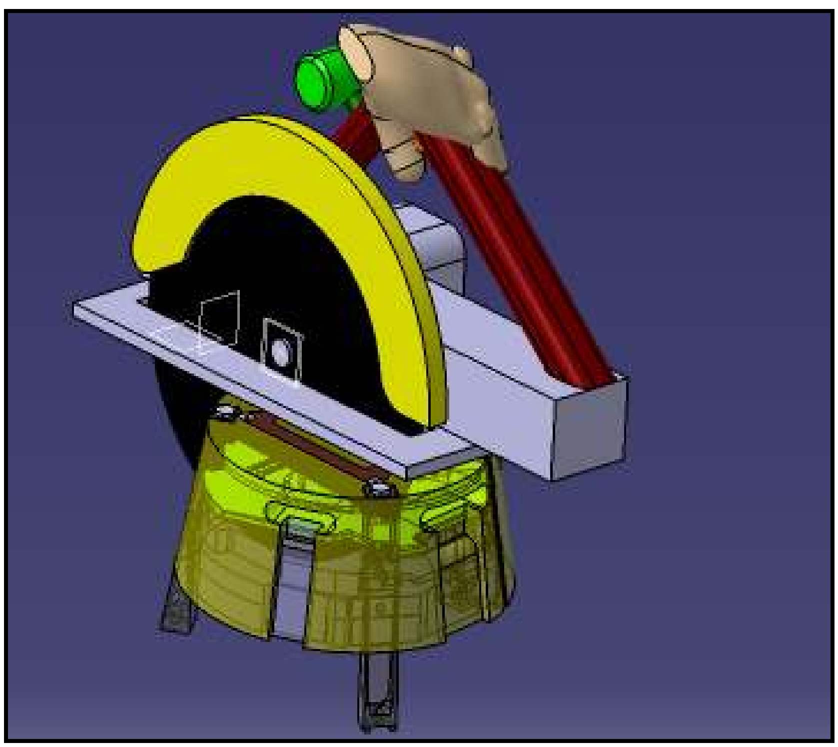


Figure 13 During cutting the assembly

5. Once you are finished with the cutting, switch off the button on the glove and press the "stop simulation" button in the Simulation window (which will be still open).

   <u>Note</u>: Make sure that you press stop simulation only once or else CATIA will terminate the process immediately without giving the end result.

6. Close the simulation window and the work piece will be updated and show the cut and this cut will also update the model in two different objects or parts Figure 14. The time taken to update depends on the speed of your cutting operation.

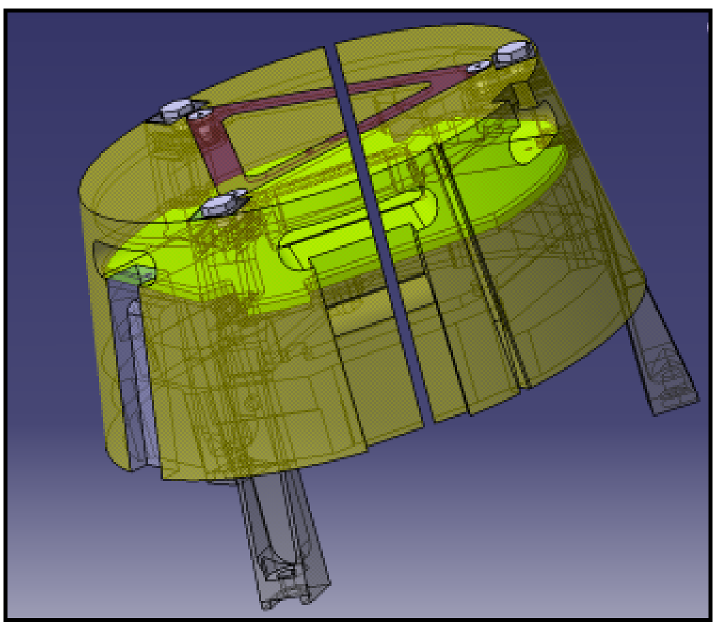


Figure 14 <u>Two halves of the foam after cutting</u>

Hit start simulation again to pick up the two separated halves and move them around independent of each other.

5.8 Capabilities

VRCut is a modified version of VRTools which will help us simulate a cutting operation. This section of the document describes the capabilities of the VRCut software.

1. The workbench VRCut in CATIA will now enable us to choose a cutting tool and start cutting any assembly selected.

2. This software now gives us the capability of cutting the workpiece when the button on the cyberglove is “Switched On” in one plane. So you can have multiple cuts in one interactive session, you just have to “Switch On” the cyberglove button when starting the cut and “Switch Off” when you want to end it.

3. Once the button on the Cyber Glove is “Switched On”, the workbench captures the instances of the cutting tool at a speed of 1 instance per 200 frames.

4. After the Assembly is cut into 2 halves it automatically updates the halves to be 2 separate parts. Once the cutting operation is over you can grab the cut parts and move them.

5. The software is flexible to allow a variety of cutting tools.

CHAPTER SIX

FORCE FEEDBACK DEVICE: PHANTOM® OMNI™

The force feedback device chosen for our research work was the PHANTOM® Omni™ haptic device from the SensAble Technologies Co Figure 15. The other haptic device which was being considered for our research was the CyberGrasp® and CyberForce® systems, from Immersion Corporation. But due to the simplicity of usage, the PHANTOM® Omni™ was selected. Although it had its own limitations, this small and simple portable device met our needs.

Figure 15 PHANTOM® Omni™

Some of the advantages of this device are portable design, cost effectiveness, compact footprint (physical area device base occupies on the desk), ease of use, the six degrees of freedom and that it can be fully customized to meet the requirements of the end user.

6.1 PHANTOM® Omni™ Technical Specification

Table 1 highlights some of the technical specifications of the haptic device used for our research.

Force feedback workspace	~6.4 W x 4.8 H x 2.8 D in. > 160 W x 120 H x 70 D mm.
Footprint (Physical area device base occupies on desk)	6 5/8 W x 8 D in. ~168 W x 203 D mm.
Weight (device only)	3 lbs. 15 oz.
Range of motion	Hand movement pivoting at wrist
Nominal position resolution	> 450 dpi. ~ 0.055 mm.
Backdrive friction	
Maximum exertable force at nominal (orthogonal arms)position	0.75 lbf. (3.3 N)
Continuous exertable force (24 hrs.)	> 0.2 lbf. (0.88 N)
Stiffness	X axis > 7.3 lbs. / in. (1.26 N / mm.) Y axis > 13.4 lbs. / in. (2.31 N / mm.) Z axis > 5.9 lbs. / in. (1.02 N / mm.)
Inertia (apparent mass at tip)	~0.101 lbm. (45 g)
Force feedback	x, y, z

Position sensing ******************************** [Stylus gimbal]	x, y, z (digital encoders) ********************************* [Pitch, roll, yaw (± 5% linearity potentiometers)
Interface	IEEE-1394 FireWire® port
Supported platforms	Intel-based PCs
OpenHaptics™ toolkit compatibility	Yes
Applications	Selected Types of Haptic Research, FreeForm® Modeling™ system, ClayTools™ system

Table 1 Specifications of PHANTOM® Omni™

6.2 Programming with PHANTOM® Omni™

This section of the chapter is divided into three sections. The first section is a needs assessment. The second one discusses the sample programs which come along with the Phantom® which would be useful with respect to our needs assessment. The last one describes the changes made to the sample programs to meet our requirements.

6.2.1 Needs assessment

As mentioned in earlier chapters our final goal is to get a full haptic force feedback model for the cutting operation in VR. This section talks about recognizing our needs and requirements to meet our final goal.

The needs and requirements are as follows:

1. Continuous vibration during the cutting operation.

2. A constant force resisting the forward cutting motion of the tool.

3. A constant and steady force in directions normal to the cutting direction.

4. Zero resisting forces when the tool is not moving forward.

1. Continuous vibration during the cutting operation

In a typical cutting operation in real life the cutting tool vibrates continuously once switched on and the amplitude of the vibration changes according to the thickness and the material properties of the material being cut. Similarly we expect that there should be a similar kind of continuous vibration when the cutting tool is switched on and also during the cutting process which might give a realistic feel of the cutting tool to the end user in our VR application. We also want this vibration to be restricted in the vertical direction only.

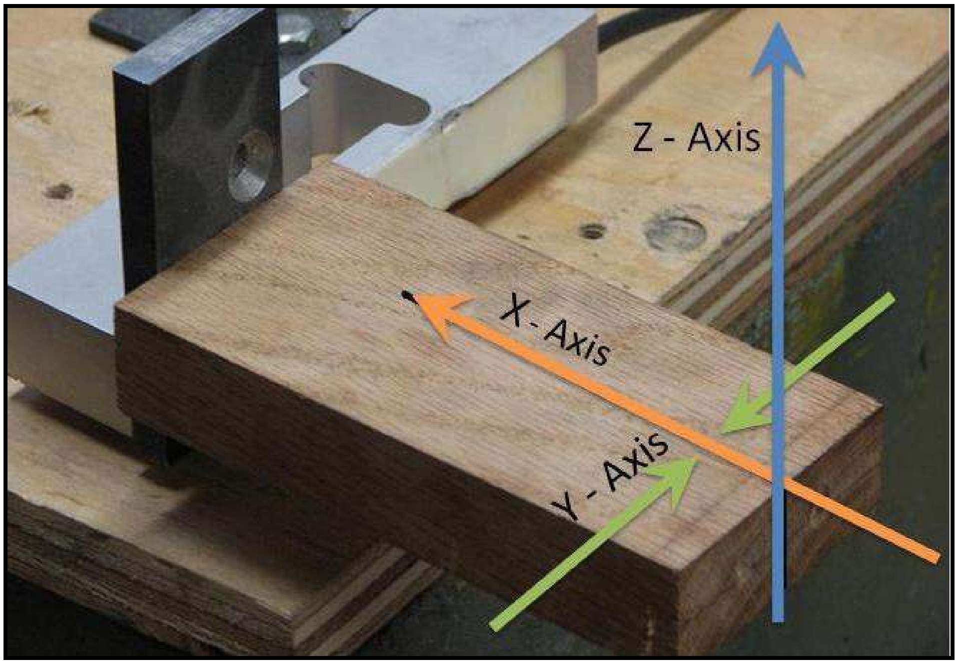


Figure 16 **Vibration in Z direction only**

As seen in the Figure 16, The X–axis is the direction of the cut. Hence we wish to have vibrations only in Z–axis direction.

2. A variable force resisting the forward cutting motion of the tool

Along with the vibration we plan to have a variable force which will resist the forward motion of the cutting tool. This force will vary according to material properties of the materials being cut, thickness or area of contact of the work piece and the cutting tool and horizontal or linear velocity at which cut is being made, i.e. as the velocity increases the force will increase gradually. Figure 17 shows the direction of the resisting force i.e. the force in negative X–axis direction which resists the motion of the tool in the positive X-axis direction.

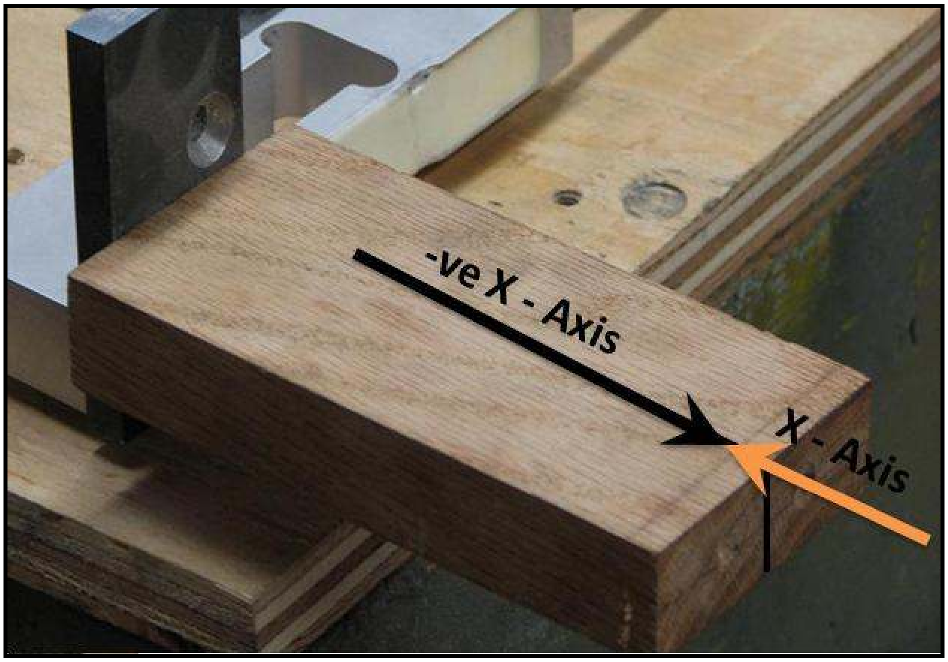


Figure 17 <u>**A variable force in negative X direction resisting the motion of cut**</u>

3. **Constant and steady force in directions normal to the cutting direction**

During the cutting process we want a steady force in horizontal directions perpendicular to the cut. This is to keep the cutting blade moving in only one direction and resist it from moving from the straight line of cut. The cutting blade will have the freedom to vibrate up and down and move forward in the direction of the cut.

Figure 18 shows if the direction of the cut is X-axis, then there should be a steady resisting force in the Y-axis from both sides to keep the motion to X-Z plane only.

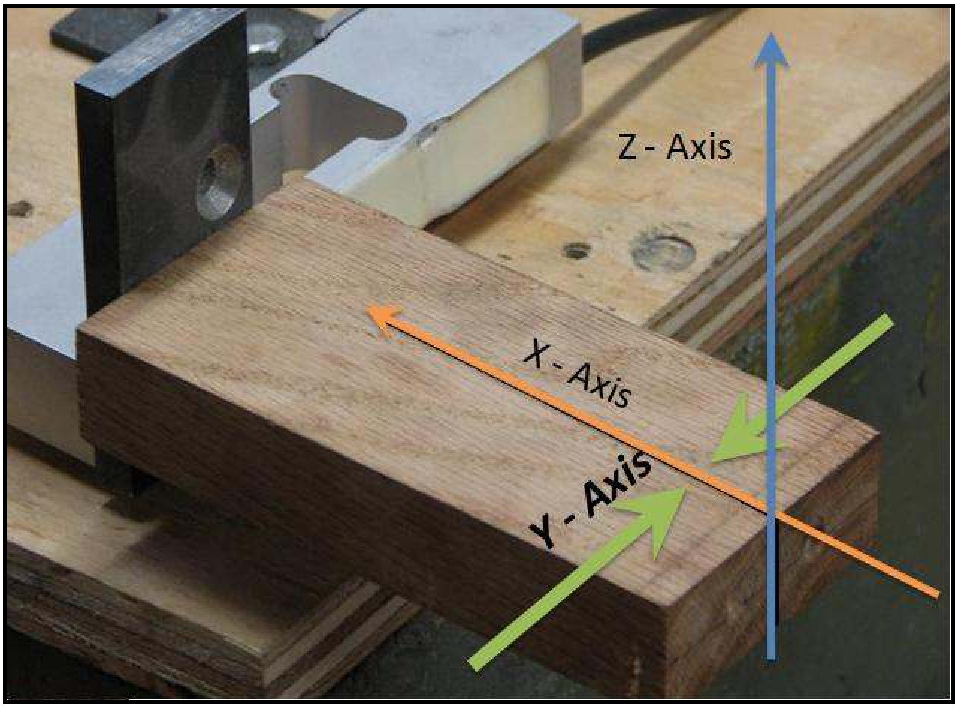


Figure 18 **A steady resisting force in the Y direction**

4. Zero resisting forces when the tool is not moving forward

We also want to model our forces in such a way that when the tool is not moving the resisting forces should be zero. This means when we are not pushing against the work piece there should be no resisting force. Also the tool should acquire this new position as its base position for further cutting.

These are some of the requirements which we expect before modeling our force feedback system. Figure 19 shows a flowchart of the process we expect in our force feedback system.

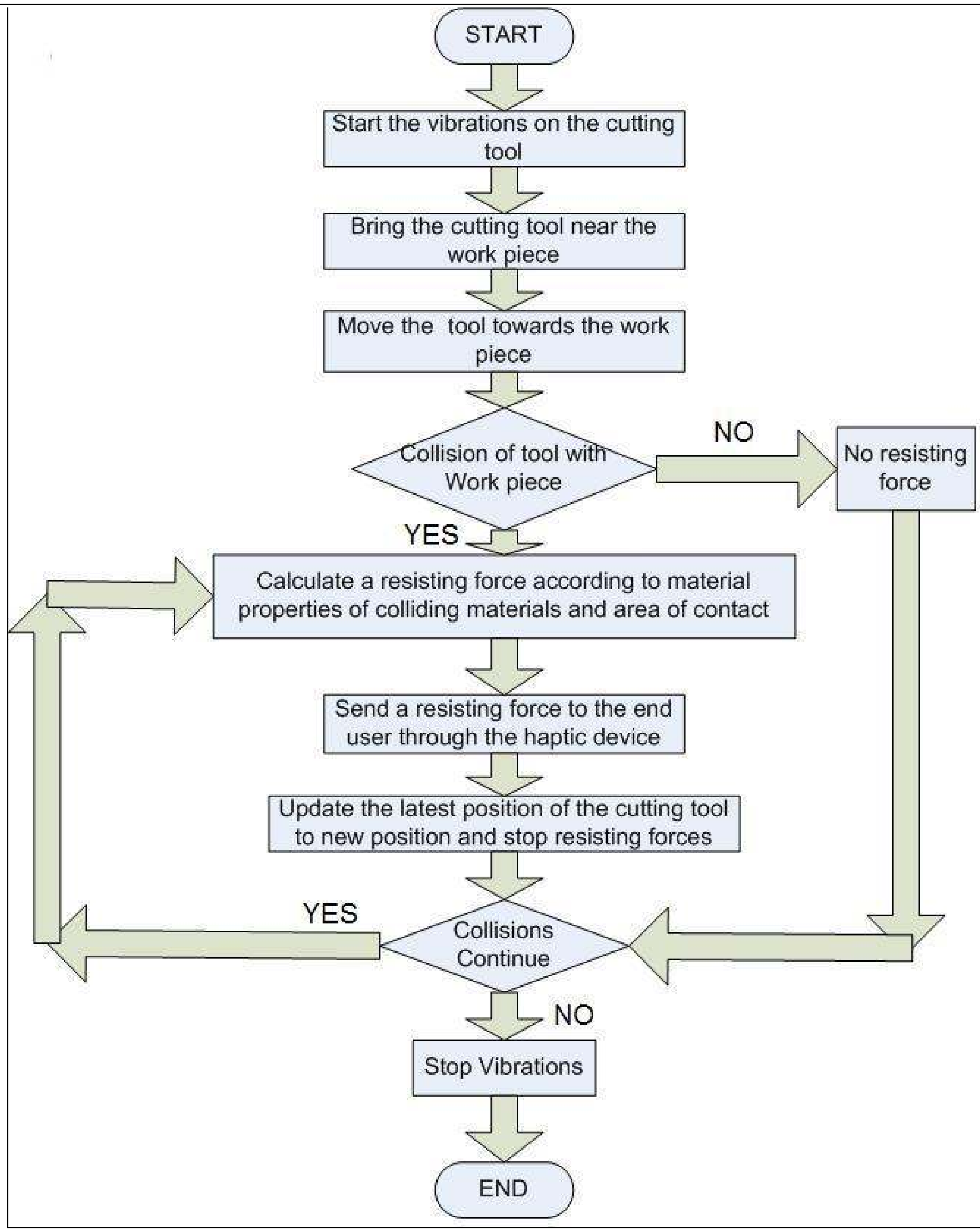


Figure 19 **Flowchart of the force feedback process**

6.2.2 Sample programs with PHANTOM® Omni™

Along with the PHANTOM® Omni™ device there were several sample programs. These programs are broadly classified as HDAPI and HLAPI. HDAPI is a haptic device API and its typical use is to initialize the device, initialize the scheduler, start the scheduler, and perform some haptic commands using the scheduler and exit when done. A HLAPI is a high level C API for haptic rendering patterned after the OpenGL API for graphics rendering. The HLAPI allows the user to specify geometric primitives such as triangles, lines and points along with haptic material properties such as stiffness and friction. The haptic rendering engine uses all this information along with the data read from the haptic device and calculates appropriate forces to send to the haptic device. The HLAPI also allows the user to set callback functions which the rendering engine will call whenever certain events, such as touching a shape or pressing the stylus button on the haptic device, occur.

After going through the entire set of sample programs, on the basis of our needs and requirements some of them were shortlisted. From HLAPI we chose three sample programs. A brief description of each sample program is given below. Here we mainly talk about the functionalities of each program and how they work.

1. **CustomForceEffect.exe:-** The effect in this example is an inertia effect, which simulates a point mass being dragged around the haptic device. Here the point mass can be changed and used for different forces. Some of the other variables which can be changed in the program are velocity, stiffness & damping constant. The damping constant is calculated based on the value of point mass and velocity.

2. **EffectAttribute.exe:-** In this particular example the end user can have various effects and in which effect magnitude and effect gain of each effect can be changed by the end user. The effects highlighted in this program are friction effect, spring effect, constant force effect and viscous effect. Once this program executes and runs, the end user can switch between the types of effects. As mentioned before the magnitude and gain can be increased using "+", "-"keys for magnitude and "[", "]" for gain increase and decrease.

3. **CannedForceEffect.exe:-** This example demonstrates starting and stopping built-in force effects for HLAPI. Force effects can either persist using hlStartEffect and hlStopEffect to control their duration or they can be triggered and run for pre-defined duration using hlTriggerEffect. In this example the magnitude and gain are set initially so the end user doesn't have the freedom to increase or decrease them. In this case when the program is executed the end user has the option of using front and the back stylus buttons for achieving spring effect and triggering effect respectively. If the

end user just moves around the stylus without pressing any button the user can feel the ambient stick-slip friction effect.

From the HDAPI we chose the following programs:

1. **Vibration.exe:-** This example shows how to generate a simple sinusoidal vibration effect for the haptic device. The amplitude and the frequency of this vibration effect can be adjusted by the end user. The magnitude ranges from 0 to 0.88 units and the frequency ranges from 0 to 500 units. The default magnitude and frequency are set to 0.66 and 100 units respectively.

6.2.3 Implementing the cutting force

Working with the sample programs available with the PHANTOM® OMNI™ device gave us a good idea of how the device works with respect to the programs. Also as stated in section 6.2.1 we knew exactly what we were looking for. Hence we thought of combining some of the sample programs to get the desired output from the device. This section describes the custom program which we prepared from the sample programs.

We chose vibration.exe from the sample programs to start with. As mentioned earlier this program shows how to generate a simple sinusoidal vibration effect for the haptic device. The default vibration frequency and vibration amplitude was set to 100 and 0 respectively. In our custom program we wanted to have the vibrations updating every frame with respect to location in space as well as the velocity of the stylus moving from its current position. We also wanted to implement a resisting force which would restrict the motion of the cut and would be in the direction opposite to the cut.

We declared variables called frame counter, last position and also had a function which would calculate the distance travelled from last position to the current position inside HDCallbackCode function. The frame counter was set to calculate the forces every 10 frames. We also set up a "if else loop" in the same function to generate forces if there was change in position. If there was no change in position from last to current new position then it was set to generate no force in negative Z direction which was also the direction of the cut.

The vibration was simulated using a sinusoidal wave in the y direction. Also to set the resistance in the perpendicular direction to the cut we set one more "if else loop" in which we tried to restrict the motion to 0.05 cm as it was the minimum limit which could be set. Apart from this we put a force of 2 N from both sides to restrict it from moving in those directions while cutting.

In this implementation, the vibration worked really well as well as the restricting forces. However, the resisting force did not work well because of vibrations in the position calculation from the Phantom®.

One of the sample codes we used and modified for our demo implementation is called the HelloSphere.exe. This code has a graphics background implemented with OpenGL libraries. In this particular code the end user can feel the surface of the sphere measuring 0.5 inch radius and push against the outer surface. Stiffness coefficient, damping coefficient and friction are some of the factors which can be controlled inside the code and these essentially give the feel of the surface.

In our modifications we replaced the sphere to a "cube" with 2 inch sides as shown in Figure 20. The user could still feel the walls of the cube similar to the surface of the sphere. The basic idea behind implementing the code with a cube was to have a rigid wall which could be compared to the work piece (object to be cut) and the stylus or the pen on the PHANTOM® Omni™ device to be used as the cutting tool.

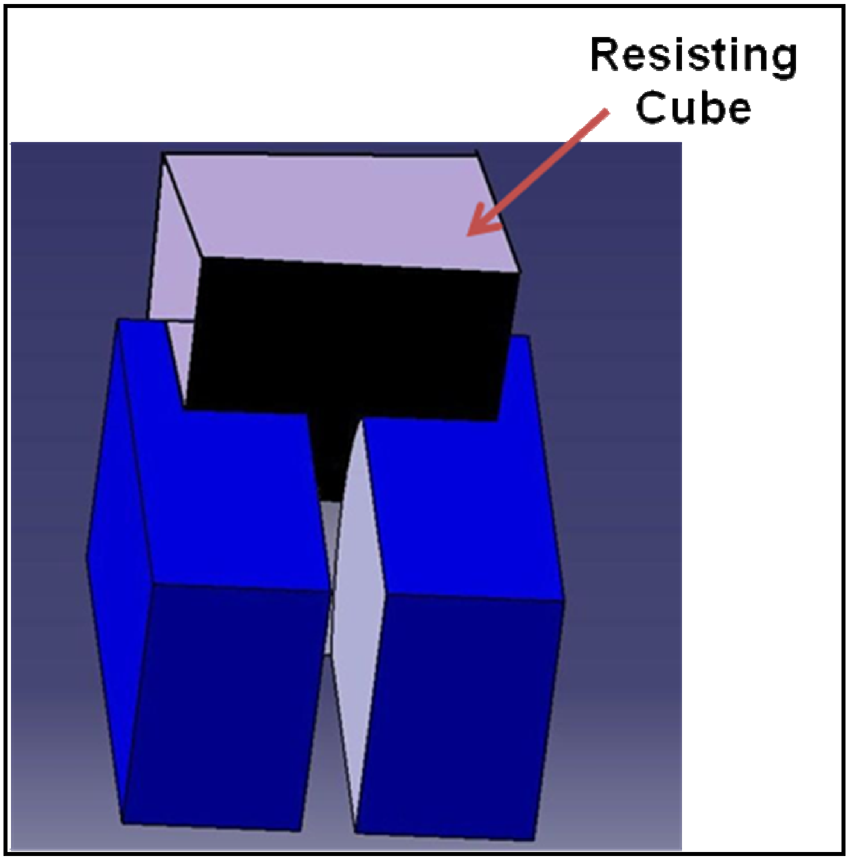


Figure 20 **A resisting cube to give the resistance to the motion of the cutting tool**

We want to have a force feedback which would be very similar to a real cutting process, and according to our needs we need a variable force resisting the forward cutting motion of the tool with the force topping when there is no forward motion. Hence we need our code to have a constant resisting motion. So when the 3D cursor which acts as the cutting tool hits the wall of the cube it gives us a resisting force according to the stiffness of the material of the cube previously set by us inside the code. This takes care of the resistance that we need. The stiffness is a variable and can change according to the material in our actual assembly in our final implementation. In our code we set the default stiffness to be 0.1 units. To show a change in material property at one specific point in the "z-direction" we set the stiffness to be 0.5 for 5 mm. Due to this the end user can feel a change in resistance as soon as the cutting tool reaches the specific point and would feel more resistance for following 5 mm.

One other functionality that we desired to achieve was, while resisting the cutting motion the cutting tool should move forward. The cube surface would be able to give us a resisting force but since the cube was stationary we couldn't move the cutting tool forward which is necessary for the simulation of the cutting process. To achieve this goal we added a function which would detect the current position of the 3D cursor and check if the tool is pushing against the wall and if it does, then the wall will move back 1 mm every frame. This movement was set only in the "z-direction". Similarly if there was no contact between the cutting tool and the cube surface the cube won't move any further.

After implementing all these functionalities and modification we were able to come up with a basic force feedback simulation. This functionality can be scaled to the required level of force feedback simulation. In this implementation we were not able to integrate the HDAPI and the HLAPI programs.

CHAPTER SEVEN

EXPERIMENTS

In order to determine the force equation to calculate forces for the force feedback system, experiments were designed and conducted on the shop floor. The basic idea behind these experiments was finding load (avg. force) vs. avg. velocity relationships for different samples of material with different material properties and different thicknesses. This would be helpful in studying the cutting behavior of the work object with respect to the cutting tool. The aim was to get results to prove the hypothesis put forward in chapter three (pg. 17). The hypothesis states that *during a cutting operation, the linear speed of cutting will increase until the cutting force equals the resistive force*.

7.1 Set up

The model used for simulation is an assembly of pieces of different material with different material properties. To take this into account it was necessary to use different materials with different material properties for the experiments on the shop floor. Different combinations of experiments were considered before designing the final experimental setup. For the ease and simplicity of the experiments it was decided to use different blocks of wood of different material properties. We used Oak, Maple, Pine and Basswood; the Oak was the strongest and the Basswood was softest among the four. A variable speed jig saw was used as the cutting tool. The multiple speeds on the saw gave more options for observing the cutting behavior at different speeds for the same material.

A data acquisition system with a single point strain gauge load cell shown in Figure 21 and a multimeter were used. The set up is as shown in Figure 22 & Figure 23.

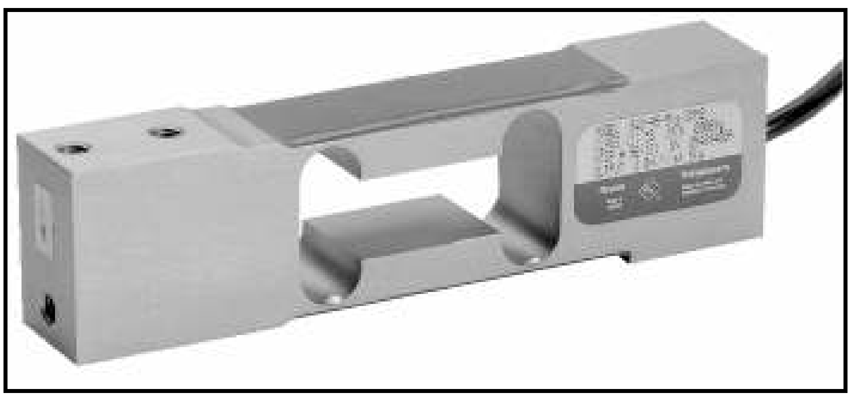


Figure 21 642C Single Point Load Cell

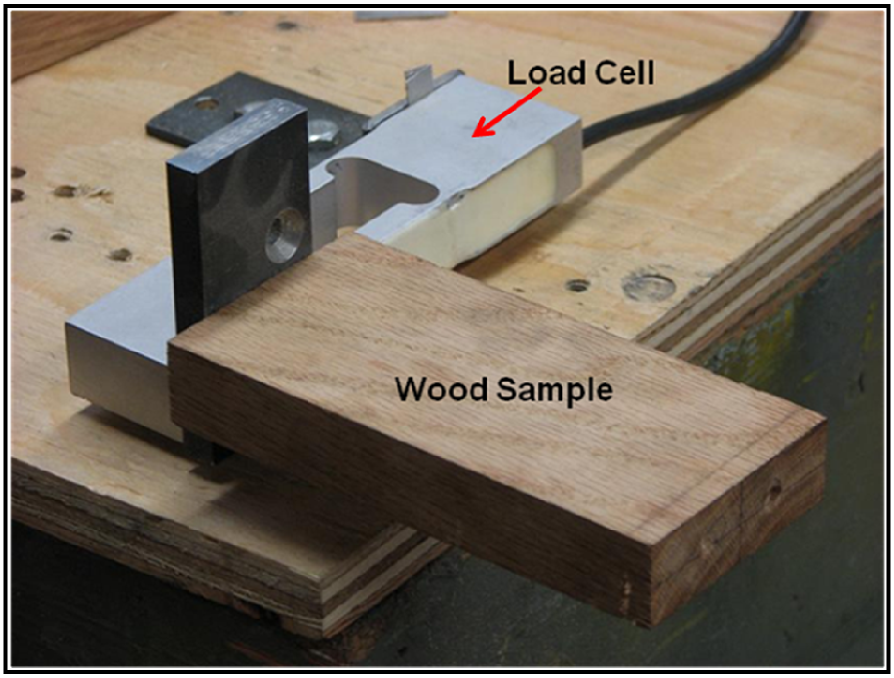


Figure 22 <u>**Final experimental set-up (a)**</u>

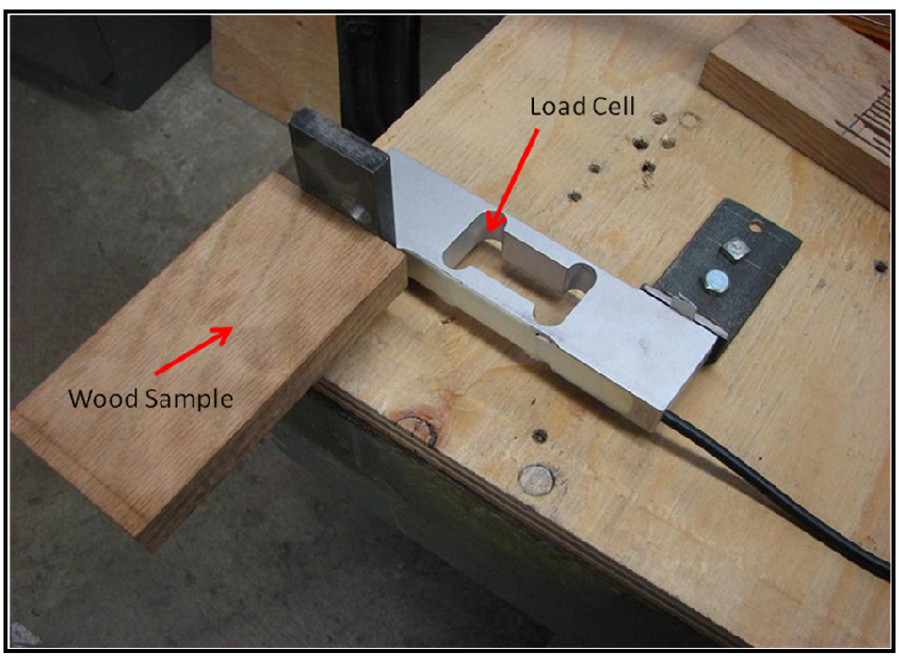


Figure 23 <u>**Final experimental set-up (b)**</u>

The load cell was connected rigidly to the base table at one end and the other end was free to access the data of the load applied. The load cell was connected to the multimeter which recorded the voltage difference readings from the load cell and sent it to an Excel Program on the computer. The spreadsheet was customized to take avg. force or load data every second during the cutting process. It also generates a graph of load vs. time which will be very useful for further analysis of forces during cutting operation. When the experimental setup was organized, we first calibrated the load cell with known weights. This gave us an equation for calculating unknown loads on the load cell which could then be used for the actual experiment. Velocity of the cut was calculated by measuring the distance of the cut which was pre marked on the wood sample and we got the time from the excel template. The wooden block was about 2″ wide. Approximately 12 cuts could be made on one wooden piece.

Figure 24 <u>Picture of the jig saw near the block just before the cut</u>

Figure 24 shows the start of the cutting process.

7.2 Procedure

The various wood pieces were cut with same dimensions so that the results from each could be compared. The detailed procedure is as follows

1. Mark the length of the cut on the wooden block. We couldn't exceed the length more than 2.5 inches (63.5 mm).

2. When everything was ready some readings should be noted on the data acquisition system without touching the load cell to calibrate the load cell at zero load or avg. force applied on the load cell. The values of the voltage readings were considered in the equation which gave us a force applied on the load cell during the cutting operation.

3. For taking accurate readings it was important to have two people for the work. One who cuts the wooden piece and the second, who starts the Excel program when the cut starts and ends it when it reaches the length of the cut.

4. Make the cuts and try to keep the thrust force constant

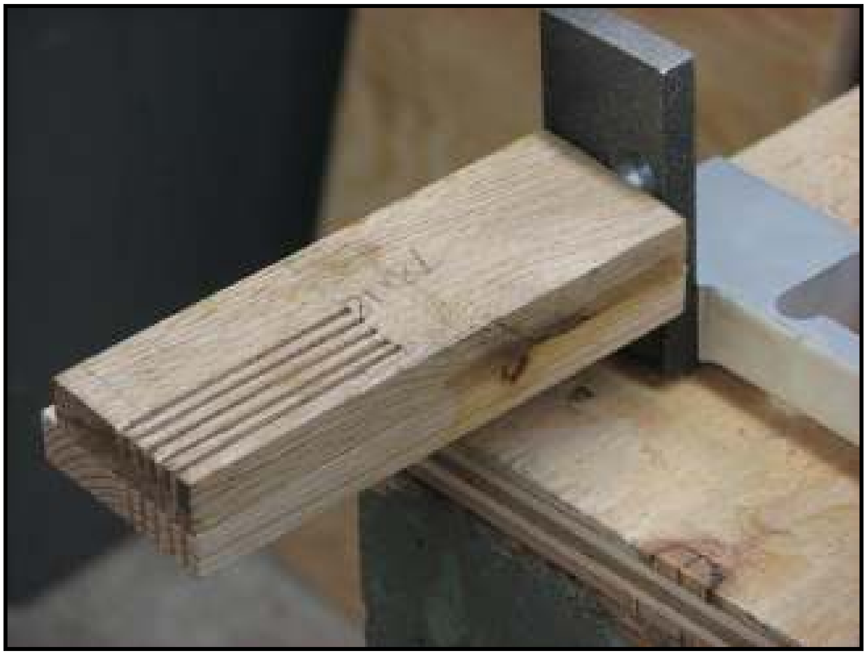


Figure 25 Picture of the already cut wooden piece

Each wooden block was cut several times and during each cut, the goal was to calculate an average velocity for a given thrust force.

7.3 Results

The results obtained from the experiments have been attached in Appendix A. In these experiments, the thicknesses of the wooden blocks were all 1″. Figure 26 below shows trends for the four types of the woods used.

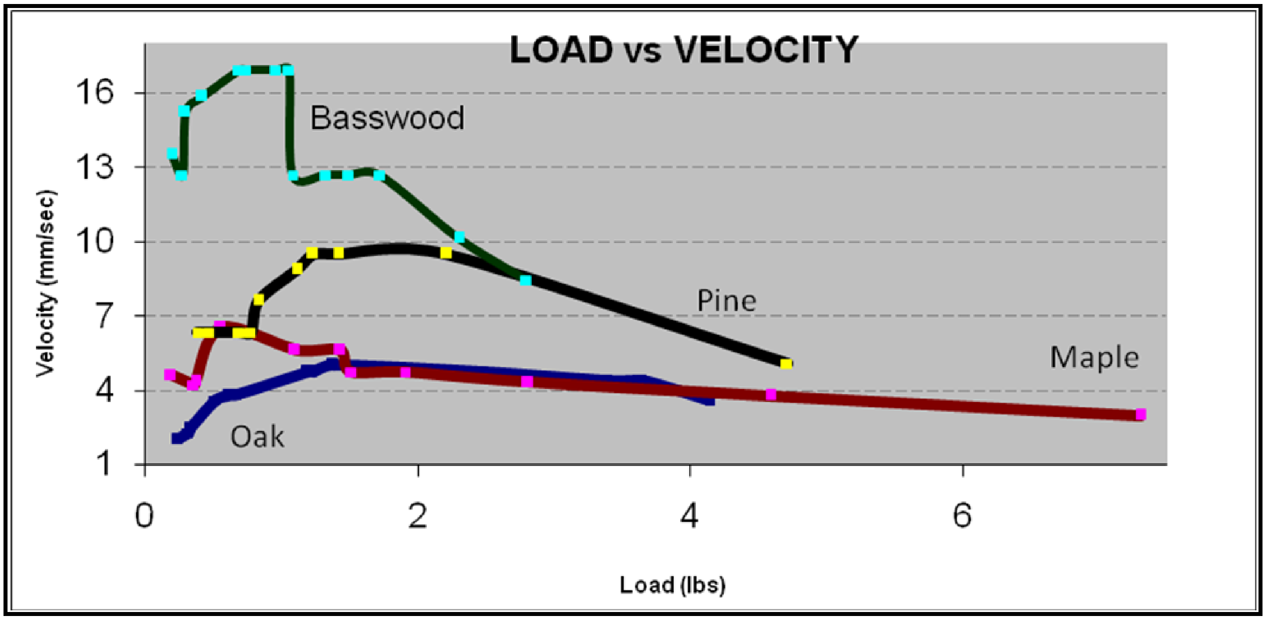


Figure 26 **Combined results for 1" Oak, Maple, Pine, and Basswood**

7.4 Discussions

The results shown above have several ambiguities. These are discussed below:

1. The excel program used for taking the load readings was set up to take readings every second. These readings could have been more accurate if it was possible to take the readings at a faster rate, i.e. around 1/10th of a second.

2. As seen in the experimental setup, the wooden block is rigidly attached to the load cell. Due to this we had considerable vibrations while cutting which directly affected the graphs in many cases. Because of this we had to carefully

review each experimental run and consider and keep only the ones that were free from the influence of vibrations for further analysis. Figure 27 shows one of the result which was ambiguous due to vibrations.

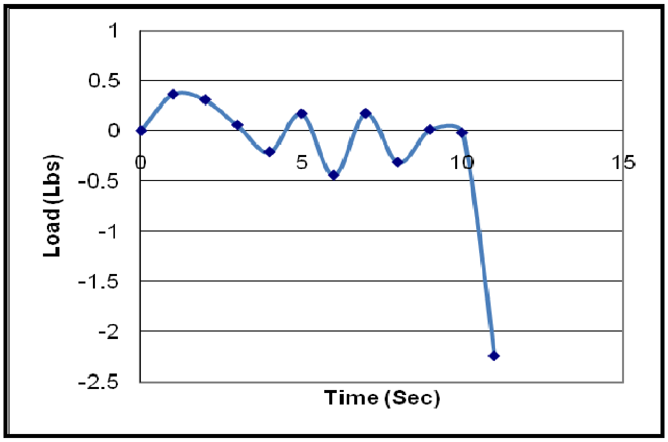


Figure 27 **Result with lot of fluctuations due to vibrations of cutting**

3. As shown in Figure 28, when the cuts were made beyond the black line on the left side the results would vary drastically. This was due to the fact that the load cell would take more accurate results at the black line because it calculates the torque generated due to the force applied by the saw on the center line.

   If we go beyond the black line on the right side, that would increase the magnitude of the torque generated. The vibrations would also increase in this case and the results would become more ambiguous.

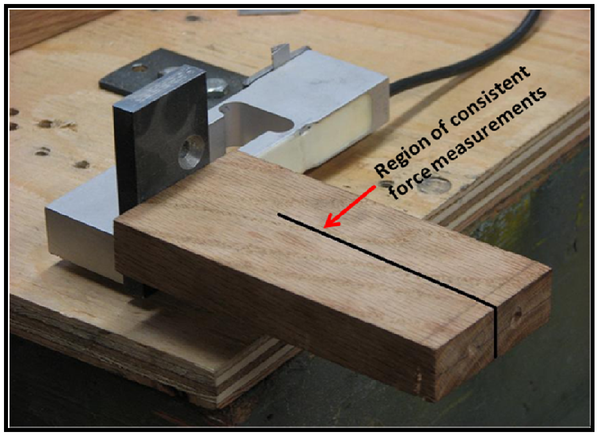


Figure 28 **Picture showing the regions where the results are inconsistent**

4. Standard deviation of the data was calculated for 10 different experiments to get a feel for the spread of the force data about the mean desired force in each experiment. The standard deviations are tabulated in Appendix B. This table shows that it is very difficult to maintain a constant steady force manually.

5. The length of the cut was marked to be 2.5 inches, but sometimes due to binding or inconsistencies in the wood grain we had to stop it before reaching the end line marked on the wooden block. This would still give us a result which could be further used.

Overall, we gained some valuable information from the experimental set–up and procedure we followed. In future we could get much better and accurate results if we can design experiments where the cutting force could be kept constant throughout a single cut, rather than manually trying to keep the cutting force constant, which is never possible.

CHAPTER EIGHT

ANALYSIS OF EXPERIMENTS

Modeling the forces for the force feedback device is one of the important issues. The data obtained from the experiments on the shop floor described in the previous chapter were used for this purpose. Every experiment with each kind of wood gave us specific results and these graphs with their respective trends were studied. The load (avg. force) vs. velocity trend for each of the wood types showed that as you increase the force, the velocity goes on increasing and as you keep on increasing this force at some point the velocity reaches its peak and begins descending because of binding between the wood sample and the cutting blade. In this chapter we intend to get a force equation which will be obtained from our load vs. velocity results from the four wood samples. This equation can be used to predict the behavior of the materials whose material properties and thicknesses are known while cutting but are different from the properties of the materials used already for experiments.

8.1 Initial Approach

Our initial approach was to make a 3D surface graph and then obtain a force equation from that graph which would predict the forces for different material properties. The experiments explained in the previous chapter were done with four wood samples with known side hardness values [18]. Hence we decided to make the 3D surface graph using load, velocity and side hardness of these wood samples. From the load (avg. force) vs. velocity behavior of different wood pieces a 3D surface graph was plotted using MATLAB. Figure 29 shows the 3D plot.

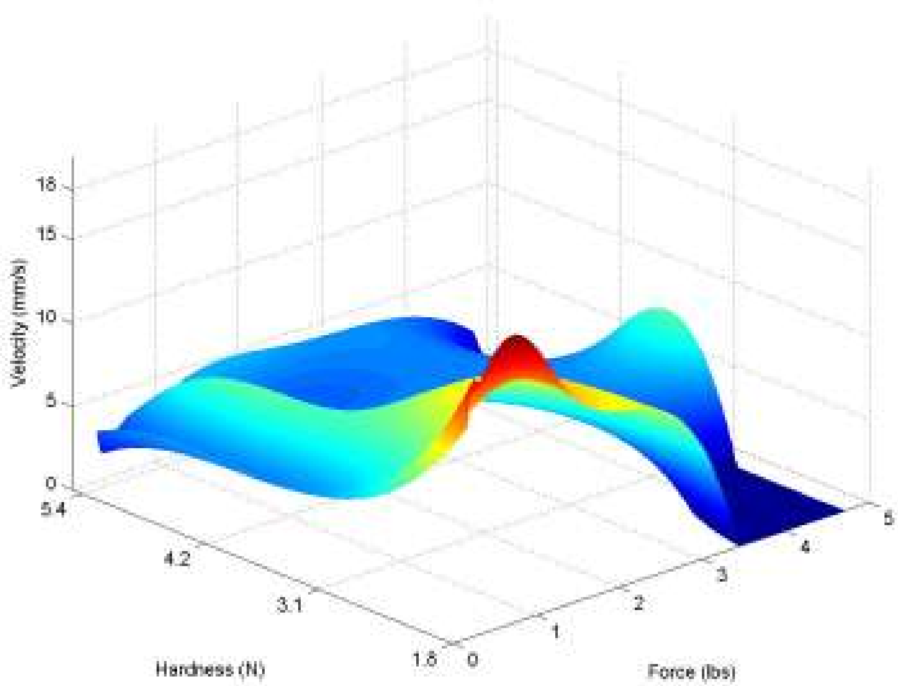


Figure 29 **<u>3D surface plot of Load vs. Velocity vs. Side hardness for the four materials</u>**

We got our desired plot, but to get a force equation from this graph in MATLAB was a challenge and we couldn't proceed any further.

8.2 Relationship of cutting velocity with respect to thickness

We conducted experiments with different thickness of the block. The goal was to understand how the load vs. velocity graph shifts with varying thickness. The cutting operation was timed for a 1″ pine and a ½″ pine (Table 2, Figure 30, & Figure 31).

No	Name	Length of cut (mm)	Time (sec)	Load or Avg. force (lbs)	Velocity (mm/sec)
1	**Pine – 1 inch thickness**	60	12	1	**5**
2	**Pine – 1/2 inch thickness**	66	12	0.52	**5.5**

Table 2 **Results from the Pine experiment**

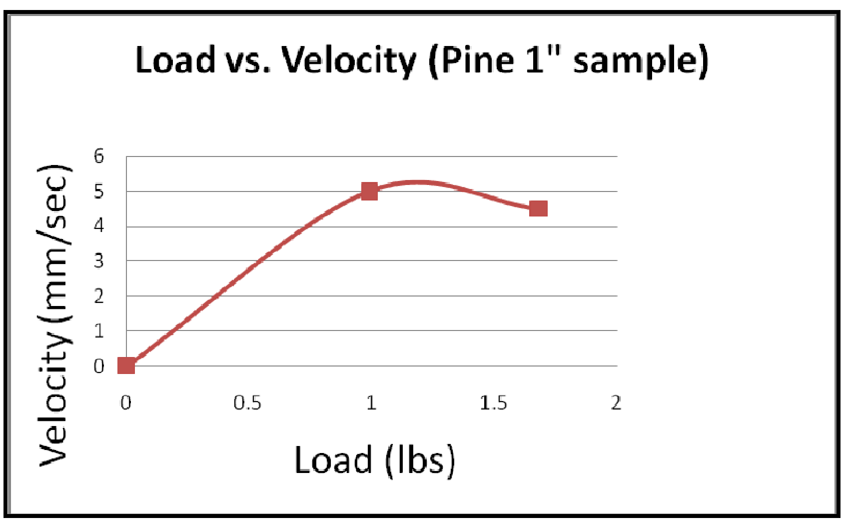


Figure 30 Load vs Velocity graph for a 1″ thick pine sample

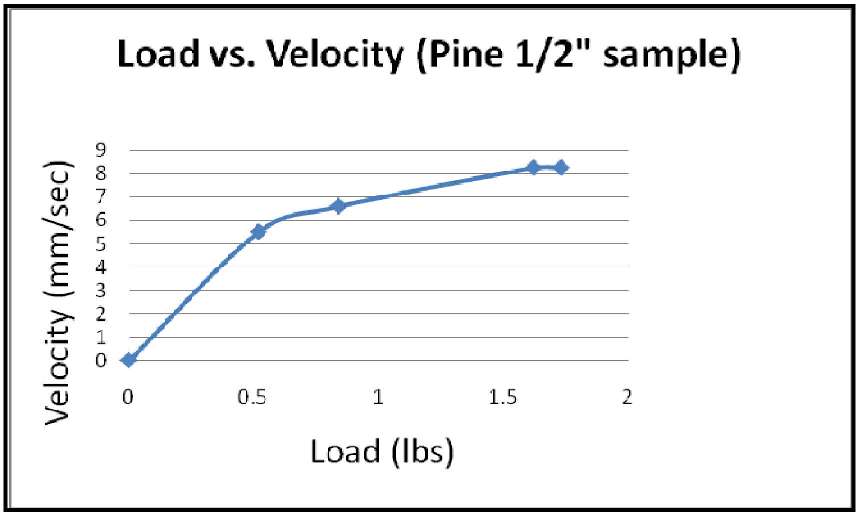


Figure 31 **Load vs Velocity graph for a ½" thick pine sample**

As seen in Figure 31, the force required to cut almost the same length is proportional to their thicknesses i.e. the force required to cut a one inch thick pine sample is 1 pound where as the force required to cut ½" pine is around ½ pound for the same distance at same velocity.

8.3 Verification of the cutting velocity behavior with respect to thickness: (PINE)

In this section we wish to verify our results from section 8.2. In this section we extended the cutting velocity behavior of the pine sample to 4 different samples of ½", 1 ", 1 ½", and 2"thickness. The results for all these four are attached in Appendix C. Our main intent in doing so was to see if the thickness vs. velocity behavior is still linear. All other dimensions other than thickness were similar for all four cases.

Figure 32 shows the load vs. velocity results for all four pine samples.

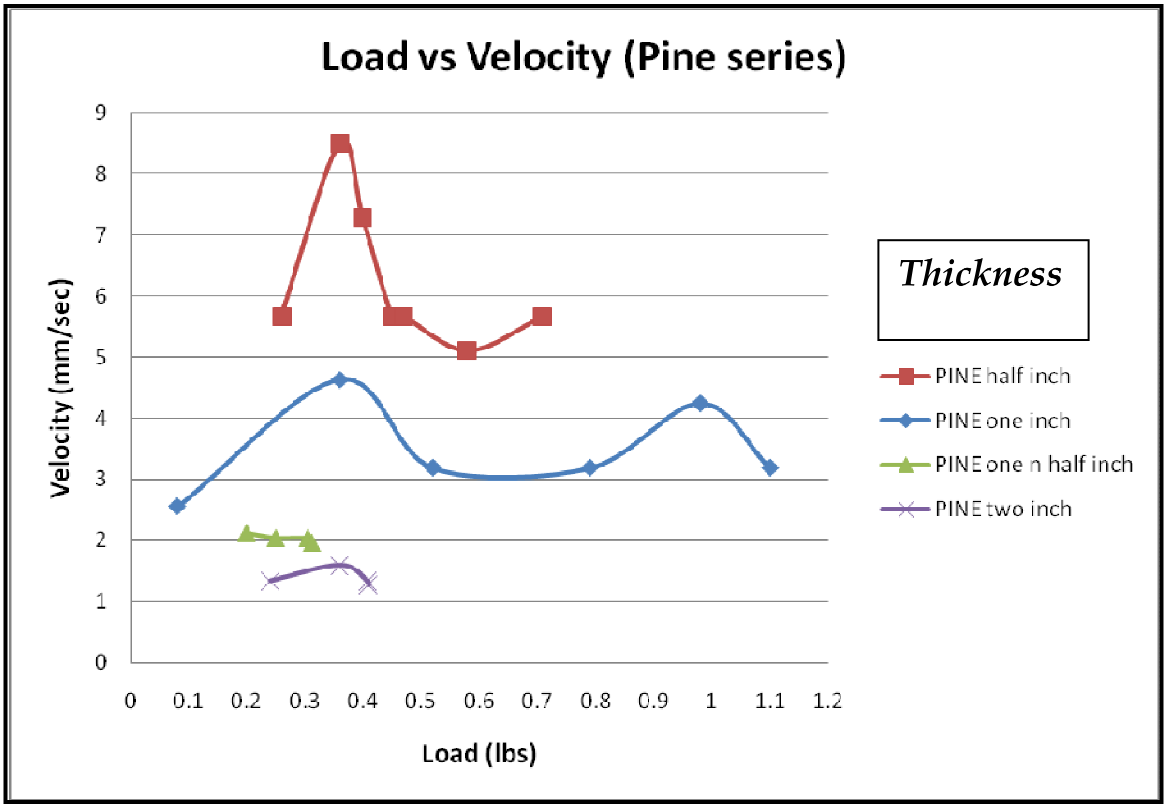


Figure 32 **<u>Load vs. Velocity plots for different sizes of pine samples</u>**

It can be clearly seen from the Figure 32 that as the thickness increases the time of cut for the same length of cut increases. Now to understand the behavior of the thickness vs. velocity at constant loads, we consider constant loads of 0.25, 0.3, and 0.35 lbs and then measure velocities of each sample at those loads for Figure 32. We tabulate these results below in Table 3.

Thickness (inch)	Velocity (mm/sec) at 0.25 lbs	Velocity (mm/sec) at 0.3 lbs	Velocity (mm/sec) at 0.35 lbs
½	7	6.8	6.6
1	3.3	3.2	3.2
1 ½	2.1	2	1.9
2	1.2	1.2	1.2

Table 3 Velocity comparison with respect to thicknesses at constant loads of 0.25, 0.3, & 0.35 lbs

From these results we plot the thickness vs. velocity graph for all three and compare them in Figure 33.

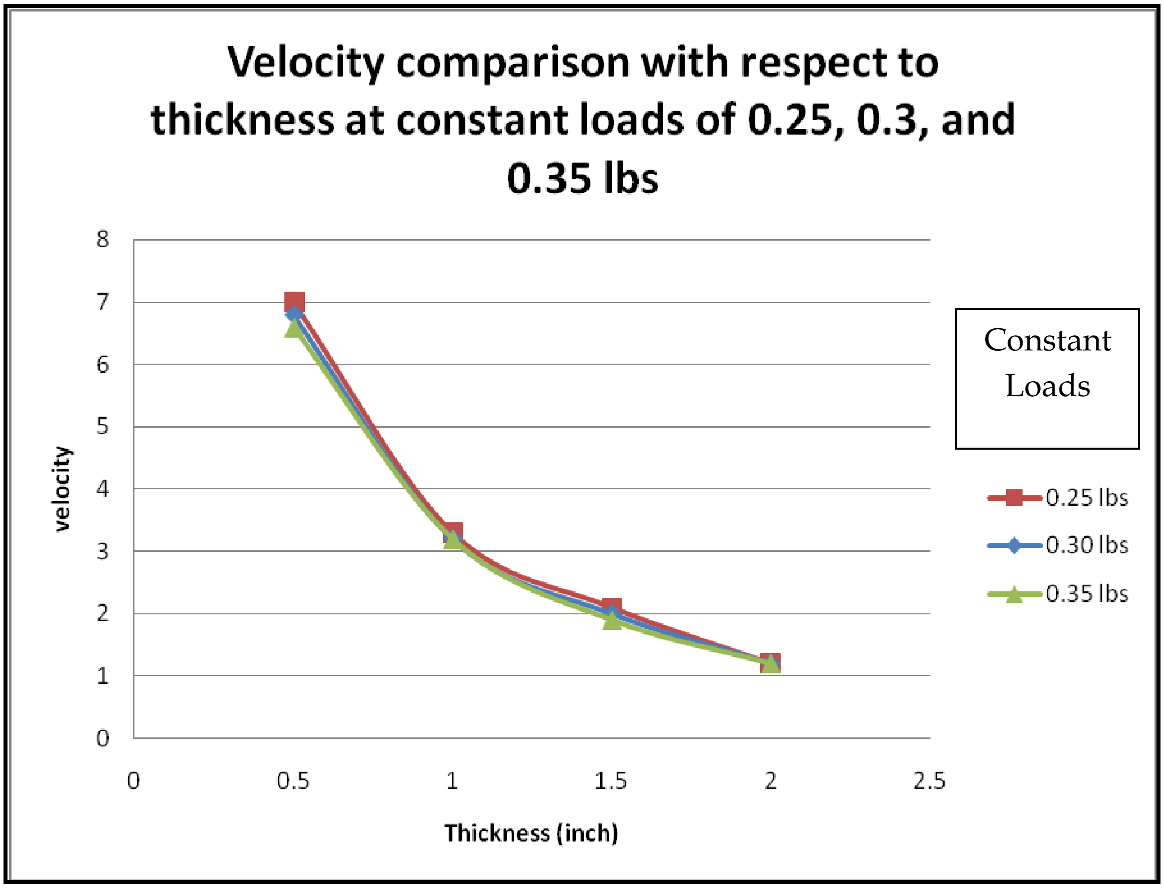


Figure 33 <u>**Velocity comparison with respect to thickness at a constant load of 0.25, 0.3, & 0.35 lbs**</u>

Here we clearly see that the velocity decreases with almost the same trend for all three results. From the graph above, it can be seen that the variation of velocity with respect to thickness is not linear. Our aim of this verification process was to understand the behavior of the cut as the thickness goes on increasing and these trends help us with the same.

8.4 Relationship between hardness and force velocity curves

From the experimental results from the four wood samples we plotted the load vs. velocity graphs and studied the trend of each of the wood samples (Appendix D). Data in Table 17, Table 18 & Table 19 represent the data from the same experiments as from the Table 8, Table 10 & Table 11 in Appendix A.

In the tables in Appendix D, only the data points that represent the initial part of the curve where the velocity increases with force are considered. This is to remove any effect of cutter blade binding with the work piece from this data. The binding of the cutter generally occurs when the cutting force exceeds the resistive force from the work piece.

In this section we try to get a mathematical relationship between the side hardness and load vs. velocity curves. It was seen that for a given hardness from our experiments that,

$$\text{Velocity} = f1\ (\text{Load});$$

The constants driving function *f1*, are related to the hardness through an equation,

$$\text{Constant} = f2\ (\text{Hardness});$$

For the functions *f1*, *f2*, we tried four different options

We plotted the graphs for these three tables, but to draw a trendline for predicting the behavior of each wood sample we considered 4 different cases. The four options which were considered for the trendline are:

1. Polynomial trendline for the *f1* and logarithmic trendline for the *f2*.
2. Polynomial trendline of degree 2 for the *f1* and polynomial trendline of degree 2 for the *f2*.
3. Logarithmic trendline for the *f1* and polynomial trendline of degree 2 for the *f2*.
4. Logarithmic trendline for the *f1* and logarithmic trendline of for the *f2*.

To explain this in further a detailed explanation of the first case is given below.

1. Polynomial trendline for the *f1* and polynomial trendline of degree 2 for the *f2*:

In this case let us consider the graphs from the different wood samples as shown in Figure 34. These graphs show the load vs. velocity behavior of the three wood samples along with their polynomial trendlines. Each graph also has an equation for the polynomial trendline. We use these equations to plot our next graphs.

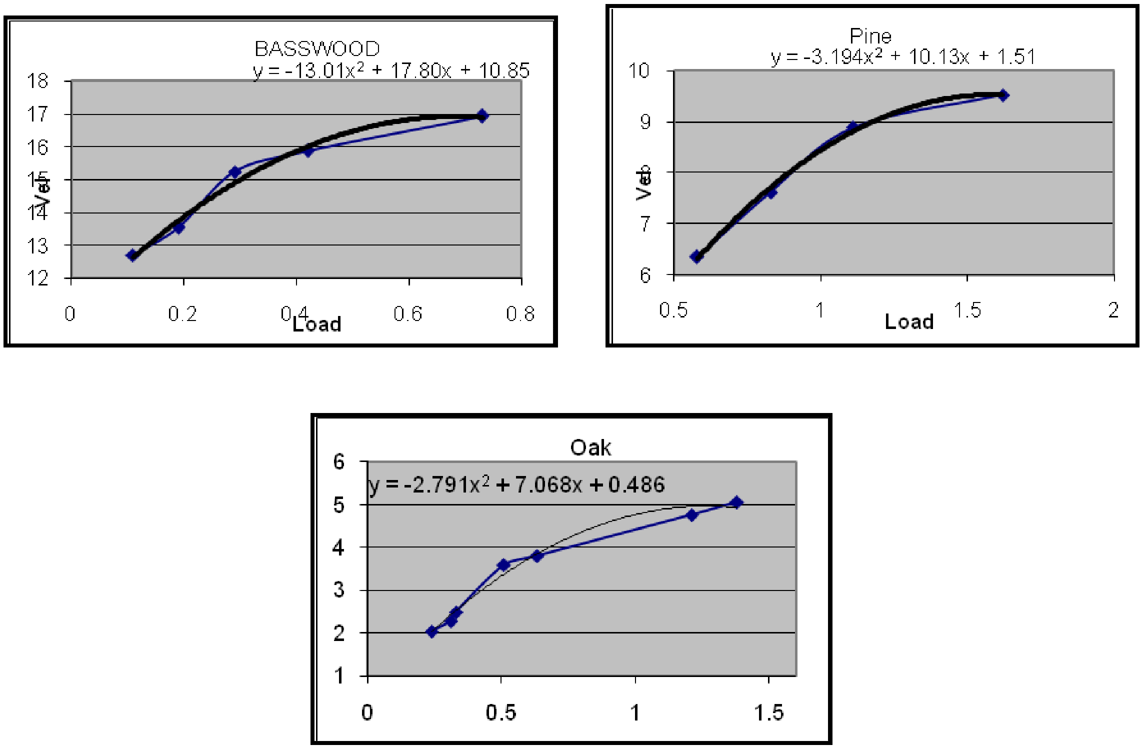


Figure 34 **Load vs Velocity graphs with a polynomial trendline for the three wood samples**

Basswood : $- 13.01\, x^2 + 17.8\, x + 0.486$

Pine : $- 3.194\, x^2 + 10.13\, x +1.51$

Oak : $- 2.791\, x^2 + 7.068\, x +1.51$

Where x = load;

We compared these equations to the format of "***ax*² + *bx* + *c***" and tabulated these equations in a table with a, b and c values. The table below shows these equations

	Side hardness (N)	a	B	c
Basswood	1800	-13	17.8	10.9
Pine	3100	-3.19	10.1	1.51
Oak	5400	-2.8	7.1	0.49

Table 4 Comparison of the values for the three wood samples obtained from Figure 34

From the table above we plotted our next set of graphs as shown in Figure 35. These graphs were side hardness vs. a, side hardness vs. b and side hardness vs. c.

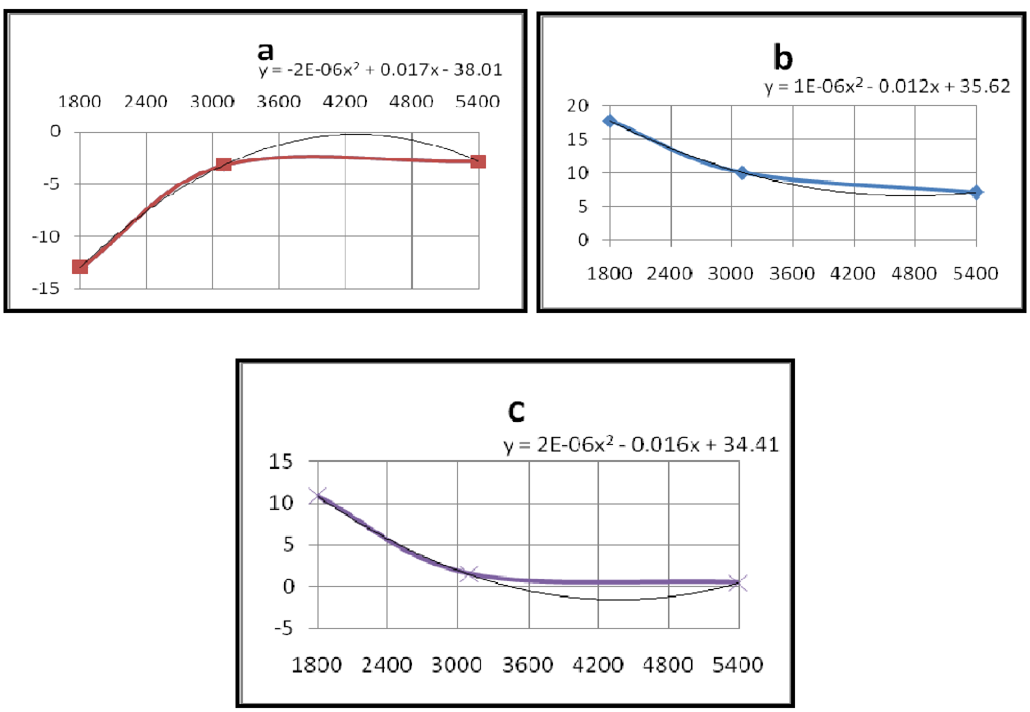


Figure 35 Plots of a,b,c vs. side hardness to obtain a trendline

The trendline which was plotted for the three graphs above was logarithmic. Our goal in doing the whole process was to see if we can predict the results or behavior of our fourth wood sample i.e. maple with a known side hardness of 4200 N. If we could get close results to what we expect for maple then further we can use these trend lines to predict the behavior of a material of known side hardness.

Hence we tried the same procedure for Maple with known side hardness. Following are the tables, graphs and results for the same.

No.	Length of cut (mm)	Time (sec)	Velocity (mm/sec)	Avg. Force (lbs)
1	50.8	12	4.233333333	0.355
2	57	13	4.384615385	0.37
3	45.72	7	6.531428571	0.5519

Table 5 **<u>Results from maple</u>**

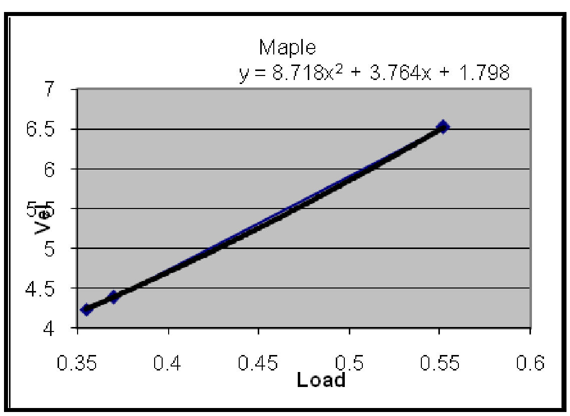


Figure 36 **<u>Trendline for maple to verify the results</u>**

Polynomial equation for Maple : $8.718\, x^2 + 3.764\, x + 1.8$; …..(where x = load)

Therefore we have

$a = 8.718$; $b = 3.764$; $c = 1.8$

but according to graphs in Figure 35 the values for a, b and c for maple should be

$a = -3.5$; $b = 8.5$; & $c = 1.5$

these results don't really corroborate our assumption.

As mentioned before we tried four cases which with different combinations of the trendline types. The one considered above was what we called "polynomial- logarithmic combination" of trendlines. Similarly we tried the other three with "polynomial-polynomial", "logarithmic- polynomial" and "logarithmic-logarithmic".

We found that "logarithmic – logarithmic" works best for us. Below are the results for the "logarithmic – logarithmic".

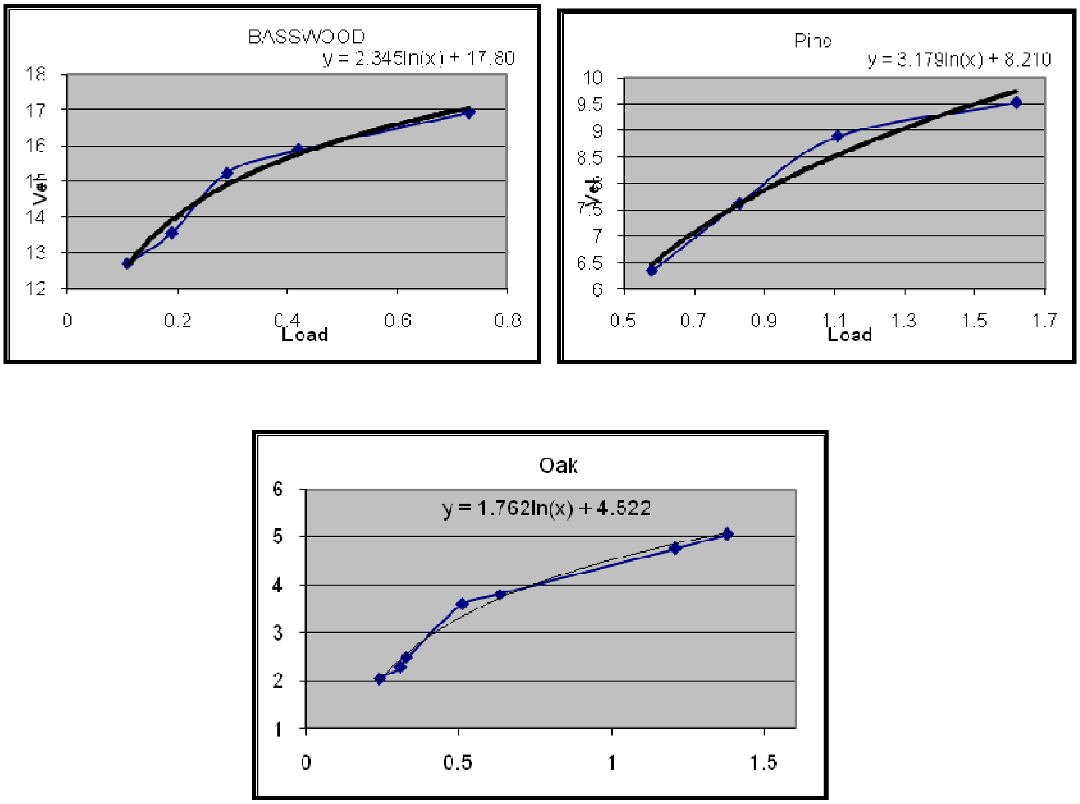


Figure 37 <u>Logarithmic – Logarithmic" results for Pine, Basswood and Oak</u>

_Logarithmic equations for the trendlines of load vs. velocity graphs as shown in Figure 37 are as follows.

Basswood: 2.345 *ln(x)* + 17.8;

Pine: 3.179 *ln(x)* + 8.21;

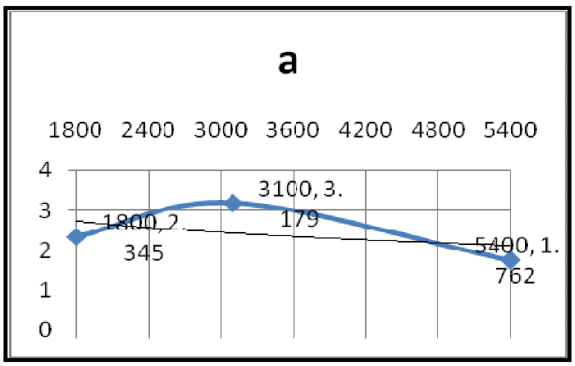
Oak: 1.762 *ln(x)* + 4.52; ……….(where x = load)

Comparing these equations with the form "*<u>a ln(x) + b</u>*", we tabulate these equations as follows.

Wood Type	Side Hardness (N)	a	b
Basswood	1800	2.345	17.8
Pine	3100	3.179	8.2
Oak	5400	1.762	4.52

Table 6 Comparison of the values for the three wood samples obtained from graph 12

We plot our next set of side hardness of the wood samples vs. a, b graphs from the above results. The graphs are as shown below.



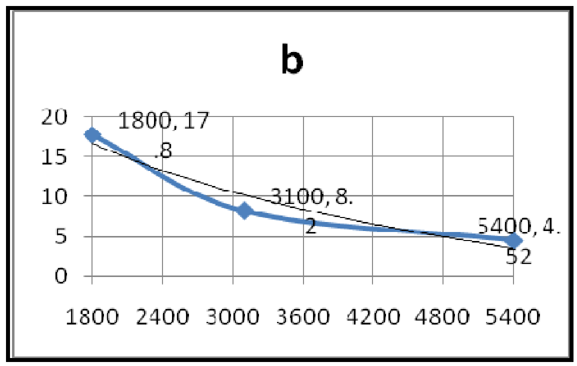


Figure 38 Plots of a,b vs. Side hardness to obtain a trendline

From the graphs in Figure 38 we predict and equation for maple with side hardness 4200 N.

Maple (4200) = 2.5 $ln(x)$ + 6.5 (prediction)

When we plot actual "logarithmic-logarithmic" graphs for Maple we got the following results.

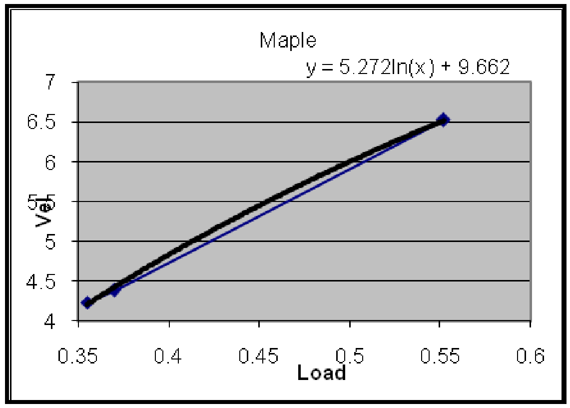


Figure 39 Vel from maple to verify the results from Figure 36

From Figure 39 graph we get a logarithmic equation from the trend line. Comparing this equation from the actual experimental results with the equation predicted.

Maple (4200): y = 5.2 *ln(x)* + 9.6 (actual from experiments)

Maple (4200): y = 2.5 *ln(x)* + 6.5 (prediction)

Where x = load & y = velocity;

This gives us the best correlation among these four methods.

8.5 Verification of cut behavior of a combination of materials

In this case we took the readings from 1″ pine sample and 1″ oak sample (results in Appendix E). From the these two results we compared them and we predicted load vs. velocity curve for a combination of half inch pine and half inch oak sample together. Once we predicted the load vs. velocity graph we compared it with the experimental load vs. velocity graph from half inch pine and half inch oak sample together.

Below are the Tables and Figures showing the same.

The side hardnesses of these materials are 3100 N and 5400 N for pine and oak respectively for predicting the results for combination. The predicted results for the combination of these two materials (½″ each) are as follows Table 7.

	Load (lbs)	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**
Side Hardness (N)		**Velocity (mm/sec)**				
3100	Experiments	4.4	4.7	4.75	4.7	4.6
5400	Experiments	2.85	3	2.85	2.6	2.6
4250	Predicted	3.625	3.85	3.8	3.65	3.6

Table 7 Predicted results for a combination of Pine and Oak samples

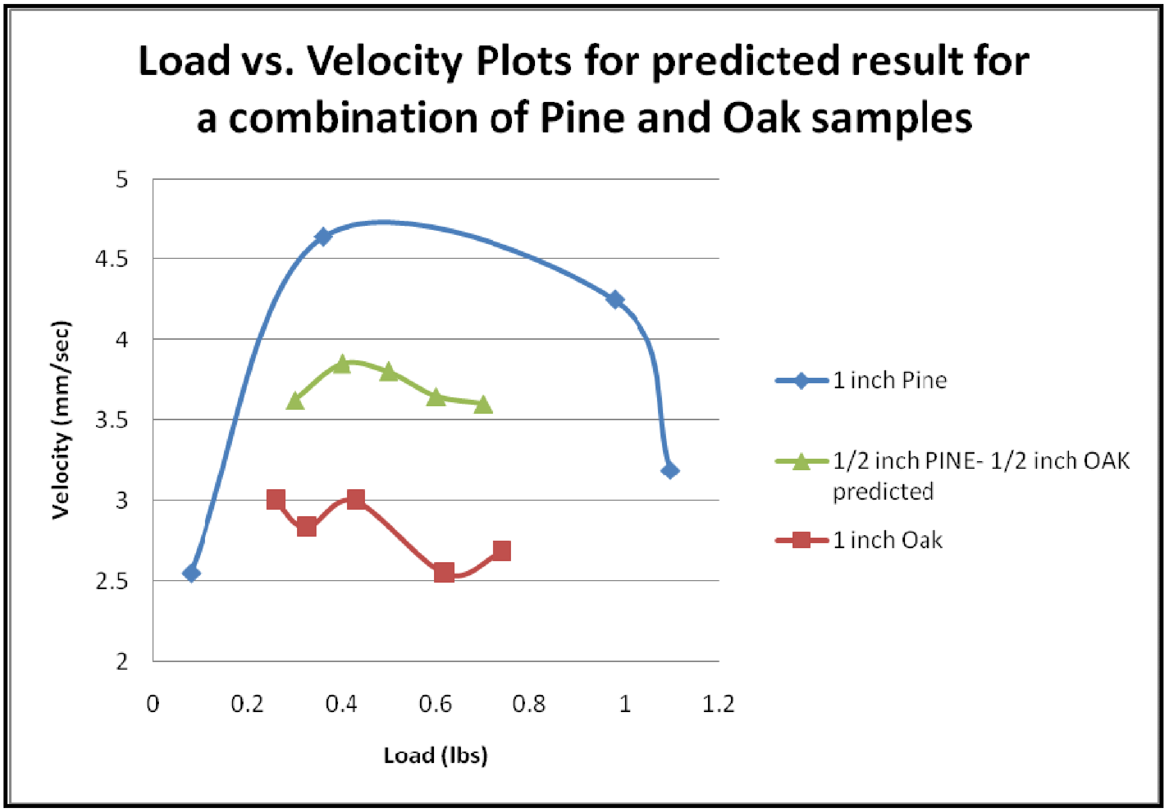


Figure 40 <u>**Load vs. Velocity plots for predicted results for a combination of Pine and Oak samples**</u>

Figure 40 shows the experimental results of 1″ Pine and Oak samples and a predicted load vs. vel for combined ½″ Pine and ½″ Oak. Next we compare this predicted result with experimental results for combined ½″ Pine and ½″ Oak. Experimental results for combined ½″ Pine and ½″ Oak are in Appendix C.3.

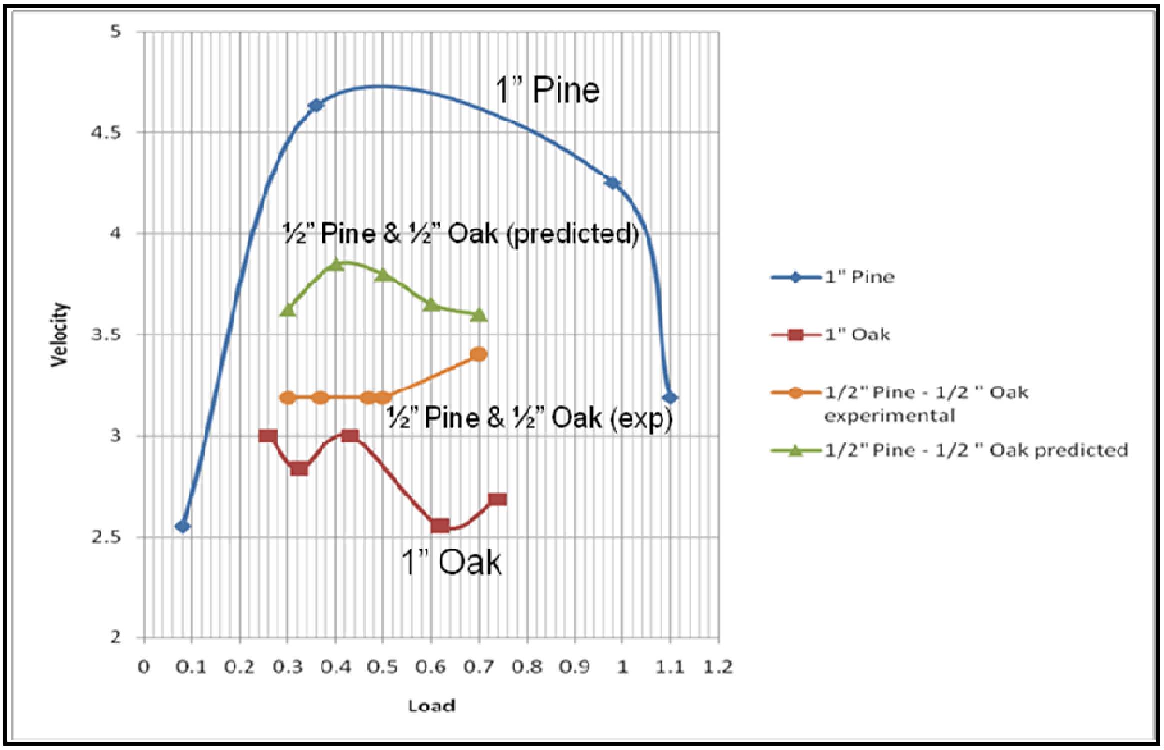


Figure 41 Comparison plot for experimental and predicted results for combined ½″ Pine and ½″ Oak samples

Figure 41 shows a comparison between the experimental and predicted results for combined ½″ Pine and ½″ Oak samples. Here we can see that the predicted results are not very close to the experimental data but still they hold good as they lie between the 1″ Pine and 1″ Oak sample which we had expected.

CHAPTER NINE

SUMMARY & CONCLUSION

9.1 Summary

The main objective of the research was to simulate a virtual cutting process integrated with CAD system and a force feedback system to provide cutting forces to the end user. We started with a literature review related to this area of research and found similar work with different approaches for varied VR applications.

We conducted experiments on the shop floor to try out the behavior of different materials of varied thickness to come up with a trend which each material follows. We extensively analyzed the results from the experiments with various combinations. This gave us a brief idea on how to model our forces for the force feedback system. Once the experiments were over, we moved to modeling the forces according to their behavior which we found from the experiments.

Our next goal was to implement the models inside a CAD system for visualization of the virtual cutting simulation and to understand methods to get the necessary information such as thickness and materials, and collisions from the CAD system interactively. We tried these methods with CAD systems like

CATIA V5 and PRO/Engineer. This led us to our proof of concept of VR CUT in which we discussed the technique of getting the information from the CAD system and coming up with a model of the virtual cutting process.

Lastly we came up with a force feedback model which explains how to model the forces and how to use the functionalities of the force feedback device. In our case we used PHANTOM® Omni™ as the haptic device. The final integration of the force feedback model with the CAD system is one of the future goals of this research.

9.2 Conclusion

We started with initial hypothesis that, "During a cutting operation, the linear speed of cutting will increase until the cutting force is equal to the resistive force". After analyzing the results of the experiments performed on the shop floor we realized that there is something more to it. As we had mentioned in our hypothesis that the force will become constant after some time, we realized that this was not completely correct. This force actually goes on reducing and goes to zero due to binding. This binding occurs when the teeth of the blade cannot remove more material from the work piece for which they are designed for.

The experiments performed on the shop floor did not have enough control due to various mechanical and human limitations and we thought they could have been much better, reliable, and accurate if these limitations weren't present. However these experiments gave us a trend to understand the forces during the cutting operation. The load vs. velocity curves achieved from the experimental data gave us a good idea of how things work during a cutting operation. The trends achieved from the experiments weren't very accurate, but they provide an approximation that might suffice for the user's feel in the virtual environment. The predicted results from the thickness verification experiments were also close to the experimental results and the trends obtained. Experiments need to be conducted in the future to compare the user's feel in the virtual environment and in the real cutting environment to validate these trends.

Our second hypothesis stated that "Typical CAD systems will be able to support the geometric modeling needs and the interaction needs of a VR – based cutting simulation". Our proof of concept of VR Cut showed that this hypothesis holds good. It had many challenges though such as working with various instances of the same model which appear multiple times in the assembly. Managing them inside the model tree after the cutting process was completed was a big challenge to us. Similarly the macro which was implemented in the program for VR Cut needs to know the names of all the models in the assembly tree along with a proper hierarchy.

The API from the CAD system was sufficient for integrating and simulating the visualization of the virtual cutting process. We had to use Macros for some parts where the API did not have adequate support. Overall the combination of API and Macros worked well in our case. Also the successful implementation VRTools and VRCut inside a CAD system corroborated our second hypothesis. Implementation of VRCut proves that geometric modeling of the cut and assembly tree reordering is feasible. However more research needs to be done for real-time calculation of material depth during the simulation, as we failed to find a way to obtain this information inside CATIA V5 or PRO/Engineer.

Overall, we were successful in deriving and implementing the various functionalities necessary for embedding a haptics-based VRCut environment within a CAD system. However significant amount of work related to verification of the force feedback models and integrating the actual haptic device within the CAD system needs to be done to implement the final product.

9.3 Future Work

In summary, some of the key items to be addressed in the future are

1. **VRCut workbench**

   Inside the VRCut workbench some key aspects which need to be addressed in the future are as follows

- The work object can be cut only in one plane. That is once it starts cutting the tool stays in one plane the moment you "Switch On" the button on the

Cyber Glove. It could be an advantageous to the user if he can cut in multiple directions in the same cut.

- The cutting tool needs to have the word Wrench in its name so that the VRTools workbench knows that it's a cutting tool and this cutting tool needs to be the first object in the model tree.
- All the parts to be cut should always be in the top level assembly. Sub assemblies cannot be cut. So that means that all the objects have to be at the same level in the model tree.
- The cut cannot be seen in real time by the user. To see the cut on the work piece you will have to press stop simulation and let the model to update to show the cut.

2. Experiments

Revising and modifying the experimental setup and procedure could play an important role in obtaining better and more accurate results. The experiments could be automated to avoid human limitations. Doing this would greatly help in keeping the force constant during the experiment to get better load (avg. force) vs. velocity plots.

3. Haptics integrated with CAD

One key factor to be taken care of in the future is the integration of a haptic system with the VRCut workbench inside a CAD system. Significant amount of work needs to be done towards this goal.

4. Integration of HL and HD API

As discussed in chapter 6, after assessing the requirements of the haptics system for the research, two different categories of API's were modified and used to get the desired force feedback. These were HLAPI and HDAPI, but using them together in a single code still remains a challenge and will be an important issue to be addressed in the future.

REFERENCES

[1] Manetta, C., and Blade, R.**, 1995, "Glossary of Virtual Reality Terminology" in the International Journal of Virtual Reality, Vol.1 Nr.2.**

[2] **Liu, G., Zhang, Y., Wang, D., Hao, J., Lu, P., and Wang, Y.,** 2005, “Cutting Force Model of Dental Training System”, Intelligent Robots and Systems (IROS 2005).

[3] **Balijepalli, A., and Kesavadas, T.**, 2003, “A Haptic based virtual grinding tool”, Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2003. HAPTICS 2003. Proceedings. 11th Symposium.

[4] **Balijepalli, A., and Kesavadas, T.,** 2002, “An Exploratory Haptic Based Robotic Path Planning and Training Tool”, Robotics and Automation. Proceedings ICRA '02. IEEE International Conference on, Volume: 1.

[5] **Tanaka, A., Hirota, K., and Kaneko, T.,** 1998, "Virtual Cutting with Force Feedback", Proc. Virtual Reality Annual International Symposium.

[6] **Yau, H., and Hsu, C.,** 2006, "Development of a Dental Training System Based on Point-based Models", Computer-Aided Design & Applications, Vol. 3, No. 6, pp 779-787

[7] **Wessi, D., and Okamura, A.**, 2004, "Haptic Rendering of Tissue Cutting with Scissors", 12th Annual Medicine Meets Virtual Reality Conference, 15-17, Newport Beach, CA.

[8] **Eriksson, M., Flemmer, H., Wikander, J.**, 2005, "A Haptic and Virtual Reality Skull Bone Surgery Simulator", World Haptics Conference, Pisa, Italy.

[9] **Mahvash, M., and Hayward, V.,** 2001, "Haptic Rendering of Cutting: A Fracture Mechanics Approach", Haptics-e, Vol.2, No.3.

[10] **Agus, M., Giachetti, A., Gobbetti, E., Zanetti, G., Zorcolo, A.,** 2002, “A haptic model of a bone-cutting burr”, IEEE Virtual Reality Conference.

[11] **Zhengyi, Y., and Yonghua, C.,** “Haptic Rendering of Milling”, Mechanical Engineering Department, University of Hong Kong, Hong Kong China.

[12] **Ove Bayard,** 2002, ”Metal cutting modeling”, Department of Production Engineering, KTH, Sweden.

[13] **Lai, W.**, 2000, “Modeling of Cutting Forces in End Milling Operations”, Journal of Science and Engineering, Vol. 3, No. 1, pp. 15-22.

[14] **Yalcin, C., Fussell, B., Erzan, Y., Jerard**, **R.,** 2004, “Real Time Calibration of Cutting Force Models for CNC Milling”, 2004 Japan – USA Symposium on Flexible Automation, Denver, Colorado.

[15] **Joshi, H.,** 2006, “Open Architecture for Embedding VR based Mechanical Tools in CAD”, Dept of Mechanical and Material Engineering, Washington State University, Pullman, WA, USA.

[16] Kalpakjian, "Manufacturing engineering and technology", Third edition, Addison Wesley.

[17] www.ie-sol.com

[18] "Wood as an engineering material. General Technical Report 113". Madison, WI: U.S. Department of Agriculture, Forest Service, Forest Products Laboratory, Chapter 4, table 4-3a, pg. 4-4 – 4-8.

Appendix A – Cutting Experiment Results and Graphs from the Four Wood Samples

A.1

Sr. No	Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
1	38.1	3	12.7	0.11
2	40.64	3	13.55	0.191
3	38.1	3	12.7	0.27
4	45.7	3	15.23	0.294
5	63.5	4	15.88	0.415
6	50.8	3	16.93	0.6905
7	45.72	2	22.86	0.709
8	45.72	2	22.86	0.72
9	50.8	3	16.93	0.74
10	63.5	4	15.88	0.783
11	50.8	3	16.93	0.958
12	50.8	3	16.93	1.055
13	63.5	5	12.7	1.08
14	38.1	3	12.7	1.323
15	50.8	4	12.7	1.49
16	50.8	4	12.7	1.72
17	50.8	5	10.16	2.3
18	50.8	6	8.466666667	2.795

Table 8 Experimental Results for 1″ Basswood

A.2

Sr. No	Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
1	50.8	11	4.618182	0.187
2	50.8	12	4.233333	0.355
3	57	13	4.384615	0.37
4	45.72	7	6.531429	0.5519
5	50.8	9	5.644444	1.098
6	50.8	9	5.644444	1.426
7	57	12	4.75	1.51
8	57	12	4.75	1.903
9	52	12	4.333333	2.81
10	57	12	4.75	2.86
11	57	15	3.8	4.59
12	50.8	17	2.988235	7.304
13	41	17	2.411765	9.59

Table 9 Experimental Results for 1″ Maple

A.3

Sr. No	Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
1	54	15	3.6	0.51
2	57	15	3.8	0.6
3	45.5	9	5.055555556	1.38
4	57	12	4.75	2.49
5	57	13	4.384615385	3.3885
6	57	12	4.75	3.511
7	57	13	4.384615385	3.644
8	54	15	3.6	4.14

Table 10 Experimental Results for 1" Oak

A.4

Sr. No	Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
1	38.1	6	6.35	0.38
2	38.1	6	6.35	0.467
3	38.1	6	6.35	0.69
4	38.1	6	6.35	0.77
5	38.1	5	7.62	0.83
6	35.56	4	8.89	1.11
7	38.1	4	9.525	1.23
8	38.1	4	9.525	1.4122
9	38.1	4	9.525	2.21
10	30.48	6	5.08	4.71

Table 11 **Experimental Results for 1″ Pine**

A.5

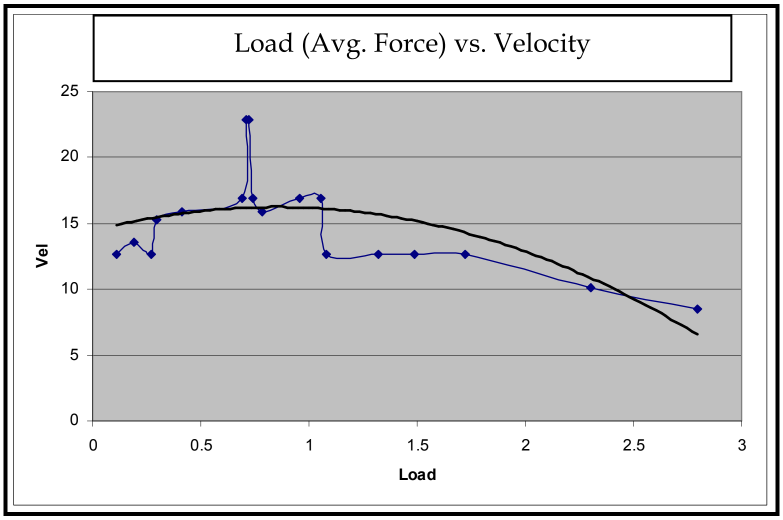


Figure 42 <u>**Load vs. Velocity (1″ Basswood)**</u>

A.6

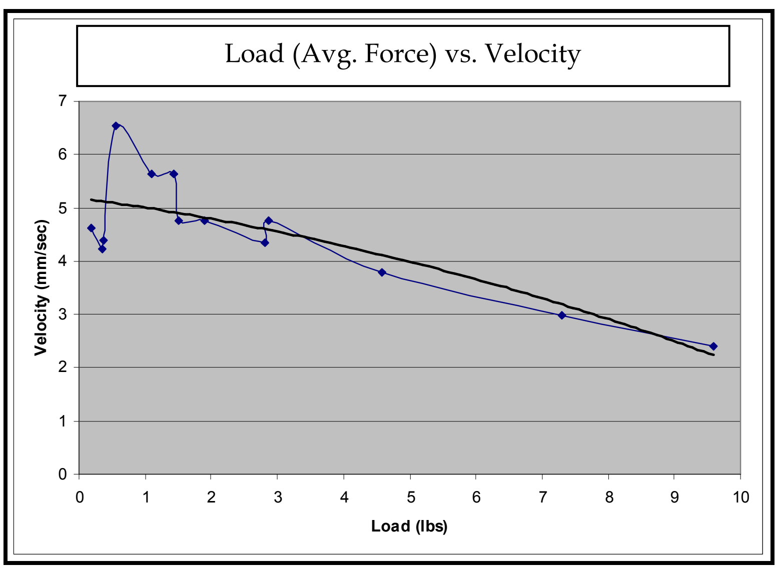


Figure 43 <u>**Load vs. Velocity (1″ Maple)**</u>

A.7

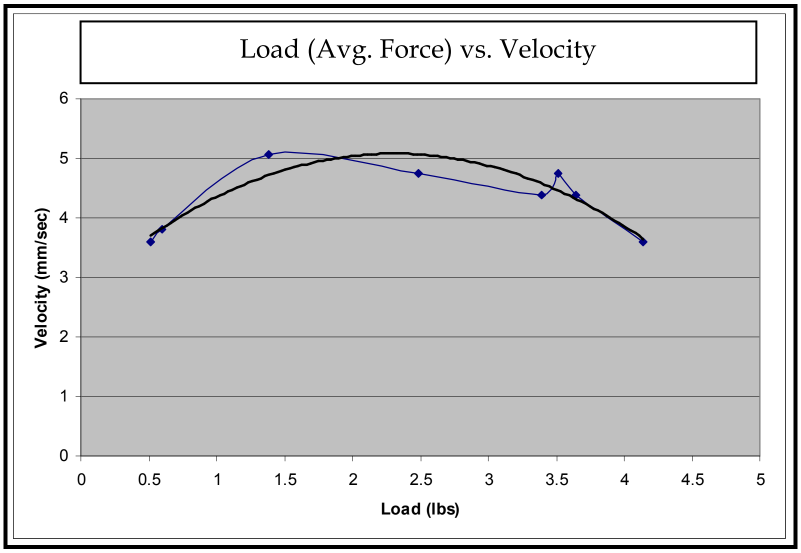


Figure 44 **Load vs. Velocity (1″ Oak)**

A.8

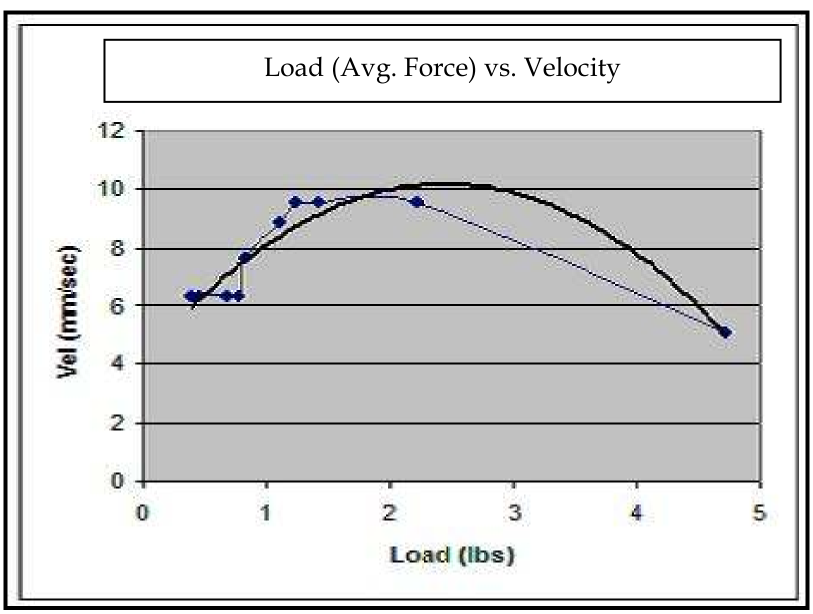


Figure 45 **Load vs. Velocity (1″ Pine)**

Appendix B – Statistical Data for Comparing Standard Deviation from 1½" Thick Pine Sample

B.1

Sr. No.	Avg. Force (lbs)	Standard Deviation (lbs)
1	0.4	0.537
2	0.83	0.14
3	0.38	0.17
4	0.77	0.4
5	0.7	0.52
6	2.21	1.3
7	1.4	0.35
8	1.11	0.48
9	4.71	1.66
10	1.23	0.84

Table 12 Standard deviation of the load during the experiment with 1 ½" Pine sample

Appendix C – Experimental Results from Pine samples of Different Thicknesses

C.1

Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
51	20	2.55	0.08
51	11	4.636364	0.36
51	16	3.1875	0.52
51	16	3.1875	0.79
51	12	4.25	0.98
51	16	3.1875	1.1

Table 13 Experimental Results from one inch Pine sample

C.2

Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
51	9	5.666667	0.26
51	6	8.5	0.36
51	7	7.285714	0.4
51	9	5.666667	0.45
51	9	5.666667	0.47
51	10	5.1	0.58
51	9	5.666667	0.71

Table 14 Experimental Results from half inch Pine sample

C.3

Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
51	24	2.125	0.2
51	25	2.04	0.25
51	25	2.04	0.305
51	26	1.961538	0.312

Table 15 Experimental Results from one and half inch Pine sample

C.4

Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
25.5	19	1.342105	0.24
25.5	16	1.59375	0.36
25.5	19	1.342105	0.408
25.5	20	1.275	0.41

Table 16 Experimental Results from two inch Pine sample

Appendix D – Experimental Results from Basswood, Oak, & Pine to Verify Cutting Behavior with respect to Hardness of Materials

D.1

No	Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
1	38.1	3	12.7	0.11
2	40.64	3	13.54666667	0.19
3	45.7	3	15.23333333	0.29
4	63.5	4	15.875	0.42
5	50.8	3	16.93333333	0.73

Table 17 Experimental Results from 1 inch Basswood

D.2

	Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
1	57.15	28	2.041071429	0.24
2	57.15	25	2.286	0.31
3	57.15	23	2.484782609	0.33
4	54	15	3.6	0.51
5	57.15	15	3.81	0.635
6	57.15	12	4.7625	1.21
7	45.5	9	5.055555556	1.38

Table 18 <u>Experimental Results from 1 inch Oak</u>

D.3

No	Length of cut (mm)	Time (sec)	Avg. Velocity (mm/sec)	Avg. Force (lbs)
1	38.1	6	6.35	0.58
2	38.1	5	7.62	0.83
3	35.56	4	8.89	1.11
4	38.1	4	9.525	1.62

Table 19 <u>Experimental Results from 1 inch Pine</u>

Appendix E - Experimental Results from Pine, Oak, & Pine-Oak Combination for Comparing Predicted and Experimental Results

E.1

Length of cut (mm)	Time (sec)	Avg. Velocity (mm)	Avg. force (lbs)
51	20	2.55	0.08
51	11	4.636364	0.36
51	12	4.25	0.98
51	16	3.1875	1.1

Table 20 **Experimental Results for one inch Pine sample**

E.2

Length of cut (mm)	Time (sec)	Avg. Velocity (mm)	Avg. force (lbs)
51	17	3	0.26
51	18	2.833333	0.325
51	17	3	0.43
51	20	2.55	0.62
51	19	2.684211	0.74

Table 21 **Experimental Results for one inch Oak sample**

E.3

Dist. (mm)	Time (sec)	Avg. Velocity (mm)	Avg. force (lbs)
51	16	3.1875	0.3
51	16	3.1875	0.37
51	16	3.1875	0.47
51	16	3.1875	0.5
51	15	3.4	0.7

Table 22 Experimental results for combined ½″ Pine and ½″ Oak samples