Computer Simulation of the Two Body Abrasive Wear Process

by

Theo Naicker

Dissertation

Submitted in Fulfilment

of the Academic Requirements

for the Degree of

Master of Science

in the

School of Geological and Computer Sciences

University of Natal, Durban

June 2002

Abstract

New computer technologies are applied to the classical material engineering two-body abrasive wear process. The computer simulation provides an interactive and visual representation of the wear process. The influence of grit size, grit tip radius and load (at constant workpiece hardness and tool path) on the wear rate, wear coefficient and wear surface topography is predicted. The simulation implements microcutting and microploughing with material displacement to the sides of the groove. The validation of the simulation is demonstrated by comparing with the previous modelling literature and with experiments.

Preface

The simulation work described in this dissertation was carried out in the Department of Computer Science, University of Natal, Durban, from January 2001 to June 2002.

These studies represent original work by the author and have not otherwise been submitted in any form for any other degree or diploma to any University. Where use has been made of the work of others it is duly acknowledged in the text.

Acknowledgments

De Beers Industrial Diamond division sponsored the work presented in this dissertation and I am grateful to them for their financial support. I would especially like to thank Roy Achilles and Martin Suckling for availing me of their hard earned engineering expertise, which allowed me to create this simulation. I would like to thank my supervisor Prof. Hugh Murrell, for his advice and guidance. I would also like to thank the two examiners of this thesis (whose names are currently unknown to me) for many useful suggestions on how to improve this thesis.

THEO NAICKER

University of Natal, Durban
June 2002

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1 Introduction

1.1 Background to the Problem Area

"*Computer simulation is the discipline of designing a model of a physical system, executing the model on a digital computer, and analysing the execution output.*" *Fishwick[11].* It is cheaper and safer to learn from mistakes made with a simulated system than to make them for real. Simulations can reduce cost, reduce risk, and improve the understanding of the system under study. Computer simulations may be applied to a wide range of fields. In the case, of the Two-Body Abrasive Wear (TBAW) process, the simulation increases the basic understanding of the wear process.

"The Two-Body Abrasive Wear process can simply be stated as the movement of one solid body over another. This is a difficult problem with a number of variables influencing the wear process. Because of the geometric complexities inherent in three-dimensional plasticity, no complete analytical model of ploughing a flat surface by an indenter with the comparatively simple shape of either a cone or a pyramid has previously been produced." Williams [3]

One of the benefits of building a computer simulation is that no matter how complicated the process, a simulation can be implemented if the domain of the process is known. This provides a good reason for studying the simulation of the TBAW process. Current numerical and statistical models of the TBAW process are often impossible to extend, because of the complicated mathematics. Computer simulations, however, are more easily extended owing to the Object Orientated design.

1.2 Goal and Objectives of the Research

Creating a computer simulation of the TBAW process can have a number of different goals. The goal of this research is to investigate the TBAW process through a computer simulation. The task then is to accomplish the following objectives:

- Review the current literature on Computer Simulations and the TBAW process, in particular that which relates to the polishing process.
- Implement a Computer Simulation of the Two-Body Abrasive Wear process.
- Investigate the different parameters of the TBAW process.
- Compare the performance of the simulation developed with the results of other researchers as well as with experimental results.

1.3 Scope and Limitations

Computer simulations of the TBAW process have been studied in the past decades. In this work a new simulation of the TBAW process is developed that takes advantage of the recent developments in computer science such as: Windows programming, object oriented programming, 3D graphics and modelling libraries. This study assumes that material is only removed from one of the bodies in the TBAW process.

1.4 Significance

The TBAW process has practical importance in grinding and polishing. In either of these applications material is removed rapidly and in a controlled manner. The TBAW simulation produces a 3D representation of the wearing process.

The significance of this research is that it demonstrates the method of applying a computer simulation to the TBAW process. The simulation shows the different TBAW performances that can be expected, and it corroborates the results obtained from traditional TBAW models. It is also significant because the simulation can be extended to cater for more complex and difficult processes.

1.5 Research Methodology

The prototypical steps in building a simulation of a complex system might be described as follows, *Sage & Olson [7]*.

- Simplify the physical system as much as possible making sure to retain the essential features.
- Identify an appropriate model of the system that represents objects that follow simple rules with specified interactions.
- Construct a computer simulation based on this model.
- Run the simulation many times with different parameters and collect the data and compare this data with the literature and experimental data .
- Identify how simple behavioural rules results in observed behaviour.

1.6 Outline of the Dissertation

The rest of this dissertation is divided into four chapters:

Chapter 2: Computer Simulation General computer science and mathematical concepts are described.

Chapter 3: Two-Body Abrasive Wear Process The TBAW process is described.

Chapter 4: Computer Simulation of the Two-Body Abrasive Wear Process This chapter surveys what a TBAW simulation entails.

Chapter 5: Experimental Methodology and Results This chapter describes the simulation runs, including techniques used, and the results of these runs.

Appendix A...C: Wear rate, wear coefficient and surface roughness error graphs.

Appendix D: Computer simulation information.

Chapter 2 Computer Simulation

2.1 Introduction

When a large, complex system has to be designed or managed, a computer program could be written to model the working of the system. The computer program can then be used to investigate expected performance, identify bottlenecks, assess design alternatives, etc. This is the essence of computer-based simulation.

"Simulation" means making a simplified representation of an original. Just as a model aircraft captures many of the important physical features of a real aircraft, so a simulation model captures important operational features of a real system. Simulation software is built from entities and processes that mimic the objects and activities in the real system.

By using mathematical methods to generate streams of pseudo-random numbers, it is possible to allow for the random variations that occur everywhere in real life. The ability to allow for randomness is one of the great strengths of simulations.

2.1.1 Simulation Importance

Consider the building of the best possible system. It is often impossible to study a system in any other way because the real system hasn't been built yet. By carefully analysing the hypothetical system with a simulation, problems can be avoided with the real thing when it is built.

Simulation studies at this stage may reveal insurmountable problems that could result in project cancellation.

We can learn from our mistakes. However, it is cheaper and safer to learn from mistakes made with a simulated system than to make them for real. Simulations can reduce cost, reduce risk, and improve the understanding of the system under study.

2.1.2 Discipline

"To characterize simulation, it is useful to compare it with other fields such as computer graphics, animation and virtual reality (VR), since these fields have much in common with simulation. Computer graphics is the computational study of light and its effect on geometric objects; the focus on graphics is to produce meaningful rendered images of the real world or hypothetical objects. Animation is the use of computer graphics to generate a sequence of frames which, when passed before the eyes very quickly, produce the illusion of continuous motion. VR is primarily focused on human-computer interaction as found in devices such as head-mounted display, position sensors and data gloves. The ultimate test of a computer animation is that "it looks good" to the viewer. Most computer simulationists, however, regard this as only one component of validation (called face validation). As long as the simulation was not for engineering or science, creating a geometric model that looks good, as it undergoes motion, is satisfactory. However, validating a mathematical model with real data (often of a non-visual nature), incorporates more than just "looks". Most VR researchers are concerned with the human-machine interaction and not with the mathematical models that actually create the artificial reality." Fishwick [13].

Working closely with people of other technical disciplines is one of the things that make simulation fascinating. This research was conducted with an industrial engineering facility, with the author having little knowledge of the two-body abrasive wear process before this research. The Simulation provides the tool to understand the physical behaviour of this process without having to run extensive physical, on site, experiments which tend to be expensive both in time and money.

"As a simulationist, the task is to understand the common vocabulary of systems, modelling terminology, and algorithmic procedures, which form the simulation foundation" Fishwick [13].

2.1.3 Representation

2.1.3.1 Models and Events

"A model is a representation of some phenomenon. The model imitates, or simulates, the phenomenon by embodying some of its properties, in one form or another. Structural models would only portray spatial, or static, characteristics and do not depict time, while models of events capture time as an essential information component. The first group of models therefore help us picture structures, whereas the second provide representations of motion of objects, human behaviour, structural changes, growth or decay. Thus we can differentiate for example, between the human figure as a structure and the human figure in motion as an event. Structural models generally portray three-dimensional objects and often are presented in three-dimensional form, though on a different scale from the objects they represent. Events cause structural models to change their shape and/or position." Pappo [5]

This research is based on a three-dimensional representation of physical objects with mathematical formulas used to simulate the objects interactions and results.

2.1.3.2 Structural Models

Copycat models of structures or objects are very common in our lives. Among the many types are: Toy Models, Life-Size Models, Artists' Models and Models of Market Products.

2.1.3.3 Structural Figurative Models

Models of structures don't have to resemble the things they represent; the information can be represented in non-iconic ways. The model does not have to be three-dimensional: Blueprints, Photographs, Maps, Math Formulas, Numerical Simulation and Event Models.

2.1.4 Evolution

"The field of computer simulation is approximately forty years old, and is still vibrant and growing. As technology develops faster hardware, old forms of simulation are made faster, and new varieties of simulation emerge through an extension process. Extending the core simulation knowledge base involves taking existing simulation concepts and blending these concepts with those outside of the simulation discipline. An example of extension is taking two concepts, a system model and an abstract programming object, and seeing how both of these relate to one another. Designing model components as objects can extend and evolve systems." Fishwick [11]

The evolution of simulations is closely related to technology advances. Before the technology age, scaled models, pictures, drawing and numerical simulations were the techniques used in Modelling and Simulation (M&S). Simulations were created with the resources available at the time. In this study a **programming object** describes each physical object of the system.

2.1.4.1 Future

"Technologies such as simulation and virtual reality will dominate the entertainment and science forefronts well into this century. With today's computer prices, personal computers are highly affordable. Armed with the computer, models of reality may be built. While what is done today may be primitive by standards set in science fiction shows such as Star Trek, the present computer simulation discipline will lead the way to these eventual goals. Digital objects may be located anywhere on the Internet and tools (or autonomous agents) would be used to locate the building block objects for the digital world. Some of this type of work is being done in Distributed Interactive Simulation which is a thrust pioneered by the Department of Defence. The implications of these types of simulations are profound since the idea of distributed simulation has enormous potential in the industrial and entertainment fields." Fishwick [13]

According to *McQuay [6]*, distributed collaboration will significantly change how modelling and simulation is employed in the 21st century organizations.

2.2 Elementary Simulation Concepts

2.2.1 Time Representation

The representation of time in a simulation is essential for some processes. Often in engineering processes time is a major factor that influences the decision-making. Unfortunately the simulation running time cannot be compared with the *real world* processes running time. The simulation speed varies on different computers (with different overall[1] speed), which affects the simulation running time.

2.2.2 Object Oriented Programming

"On one hand there is the real world, which is full of objects and interactions, and on the other there is a computer program. A central goal of computer simulations is to map one onto the other. The most straightforward way of doing this is to create objects in the programming language, where these objects map directly to real world objects. This approach was first developed in the Simula language and has gained greater momentum over the past five years. One reason for the lag in OO-based design is that no good visual analog existed for representing class hierarchies, objects and object interactions. The past five years have produced good visual OO techniques, mostly from the software engineering community." Fichwick [11]

These techniques are exploited in this work. Using a home PC the visual objects and their interaction are shown.

[1] Depending on speed of the processor, hard drive, bus, RAM and motherboard.

2.2.3 Height Map Data Structures

The speed and reliability of a simulation depends on the data structures used. This section focuses on the height map data structure (HMDS) that is used to model surfaces in this study.

For the purpose of this research all 3D objects are described by their surface topography. The use of the HMDS produces an accurate representation of the objects surface and simplifies the programming code.

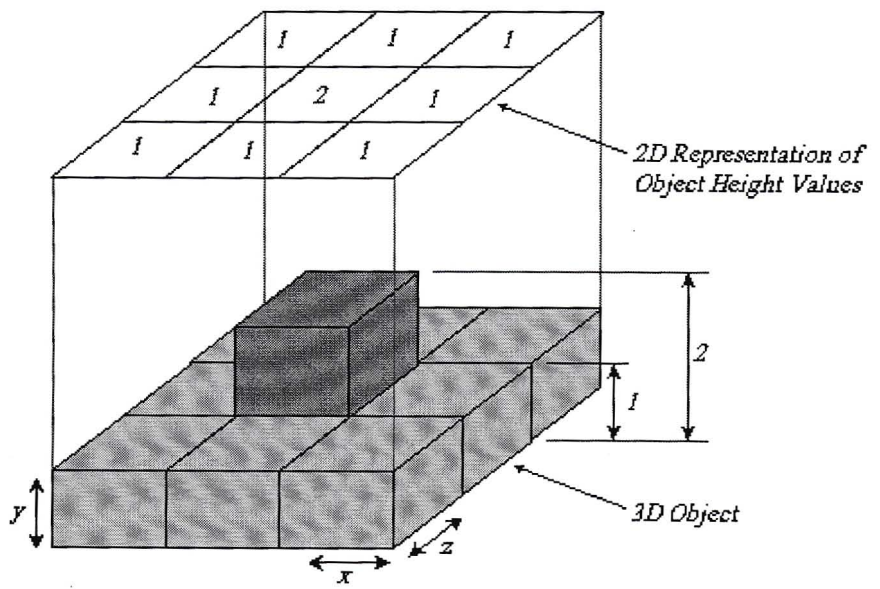


Figure 2.1 Representing a 3D object as a 2D array.

Figure 2.1 illustrates a 3D object stored in a 2D array. The object is constructed from little cubes. The array size is calculated by projecting the object's boxes onto the *xz-plane*. Each square in the array has a number to represent the height (*y-axis*) of the object at the corresponding *xz position.*

2.2.3.1 Resolution

The *resolution* is always associated with the number of pixels in a given area.

$$\text{Resolution} = \frac{\text{Total number of Pixels}}{\text{Area}} \qquad (2.1.)$$

From Equation 2.1, increasing the *total number of pixels* will result in an increase in the *resolution,* and therefore a better visual picture (Figure 2.2).

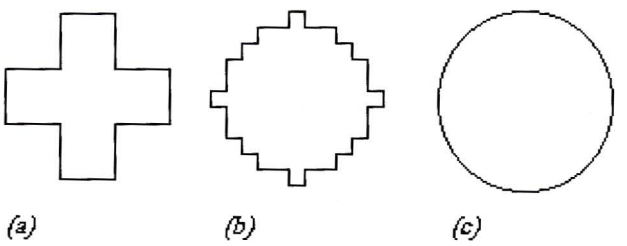


Figure 2.2 Three different resolutions representing the circle. (a) Low resolution (b) Medium resolution (c) High resolution

The 3D object in Figure 2.1 is drawn at low resolution. Increasing the resolution therefore increases the storage space required for the array data structure. The time required to traverse a larger array also increases. The resolution factor is taken into account when modelling an objects surface.

2.2.4 Colour Interpolation

Colour interpolation is used to smoothly shade a surface. A simple method is to assign different colours to the different height values (*y-axis*) on the objects surface, as in Algorithm 2.1. This makes the surface features easier to interpret.

Algorithm 2.1 Shading of Surface

INPUT *max* is the maximum surface height
min is the minimum surface height

point[*i*].*height* returns the surface height value at position *i*
point[*i*].*colour(R, G, B)* sets the colour at position *i* $0 \leq R, G, B \leq 1$

//calculate centre position, if the difference between max and min is odd then add one to
//the position of the centre line
centre = ((*max-min*) / *2*) + ((*max-min*) mod *2*);
actual_center = *min* + *center*

//traverse all surface points
FOR *i* FROM *0* TO *All points on Surface*

IF (*point*[*i*].*height* ≤ *actual_centre*)
//Red to Green
point[*i*].*colour*((*actual_centre* – *point*[*i*].*height*)/*centre*, (*point*[*i*].*height* - *min*)/*centre*, *0*)

ELSE
//Green to Blue
point[*i*].*colour*(*0*, (*max* - *point*[*i*].*height*)/*centre*, (*point*[*i*].*height* - *actual_centre*)/*centre*)

■ End of Algorithm

Algorithm 2.1 assigns the colour *red* to the lowest surface height value, *blue* to the highest surface height value and *green* to the centre surface height value. All other surface height value colours are interpolated either between *red* and *green* or *green* and *blue*.

2.2.5 Search Methods

2.2.5.1 Linear Search

The sequential search, also known as the linear search, is the most basic search algorithm and is often the first search method used. The basic strategy is straightforward. Every element in the data set is examined in the order presented until the value being searched for is found.

Sequential search is at best O(1) , at worst O(n) , and on average O($n/2$) . If the data being searched is not sorted, then it is a relatively efficient search, [16]. However, if the data being searched is sorted, other search methods could be used.

2.2.5.2 Binary Search

Binary search is a more specialized algorithm than sequential search as it takes advantage of data that has been sorted. The underlying idea of binary search is to divide the sorted data into two halves and to examine the data at the point of the split. Since the data is sorted, ignore one half or the other depending on where the data lies in comparison to the data at the split. This makes for a much more efficient search than linear search.

Binary search is used on sorted arrays, but used more often with binary search trees. Whereas linear search allows us to look for data in O(n) time, where n is the number of elements being searched, binary search conducts the same search in O(log n) time, a dramatic speed enhancement.

Binary search is one of the most common search algorithms and is useful in most real world applications written (assuming data is sorted).

2.2.5.3 Modified Linear Search

This method uses a modified *linear search* on sorted data. The searched element may not be contained in the data set and the search does not always start at the beginning of the data set.

This method is applied when the binary and linear search methods fail in this application. If data is sorted the linear search will not be efficient. The binary search will not be able to find values close to the searched value if this searched value is not in the data set.

The characteristics of the data set used are:

- Sorted data
- Values may not be found in the data set
- The position of the current searched value is found close to the position of the previously searched value in the data set.

Algorithm 2.2 Modified Linear Search

INPUT *getValue[i]* is the sorted data set, and returns the value at position *i*.

gPos = 100 //array index at the beginning of the array

WHILE another search value *F*

{

posT = gPos //array index

*δ= 0.2*F* //Delta range

final = false //return the largest value below F - δ

xMaxBelow = 0 //records the largest value below F - δ

```
found = false;                    //stop if searched value found

WHILE (NOT found)
{
        IF (final)
                posT = posMaxBelow;
                found = true;
        ELSE
                posT = posT - 1;

        x = getValue[posT];

        IF ((x ≤ (F - δ)) AND (x ≥ xMaxBelow))
                xMaxBelow = x;            //the highest lowest value
                posMaxBelow = posT;

        //x value is in the delta range
        ELSE IF ((x ≤ (F + δ)) AND (x > (F - δ)))
                found = true;
                gPos = posT + 3;

        ELSE IF (x > (F + δ))
                IF (posMaxBelow > 0)
                        final = true;
                ELSE
                        posT += 2;      //assume the highest lowest value was not found
} //end while not found
} //end while more search values

RETURN x;
```

■ End of Algorithm

Refer to Algorithm 2.2. Given a value, *F,* to search for in the data set. If *F* is found in the data set, the algorithm stops and returns *F.* Assuming *F* is not found, the closest data set value to *F* within the range [$F + \delta$, $F - \delta$] (delta range) is chosen. Finally if *F* is still not found then the maximum data set value smaller than ($F - \delta$) is chosen.

2.3 Mathematical Concepts

2.3.1 Length of a Curve

The length of a curved line is calculated by dividing the curve into small portions (Figure 2.3). The smaller the curve portions the greater the accuracy for calculating it length.

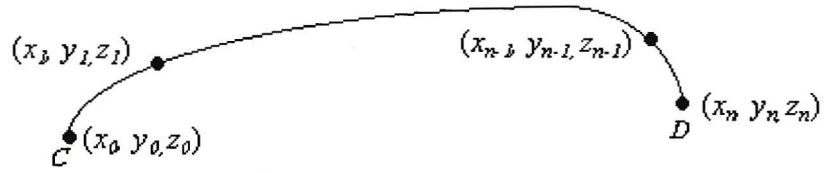


Figure 2.3 Curve length divided into equal intervals

The sum of the straight-line distance of each portion (Figure 2.3) is found by using Equation 2.2. [17]

$$\overline{CD} = \sum_{i=1}^{n} \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2} \qquad (2.2.)$$

2.3.2 Angle in the region

For the purpose of explaining the concept of *angle in the region* degree is used instead of radians. This section consist of checking if an angle β is less than $\alpha + \delta$ and greater than $\alpha - \delta$. This is not a straightforward comparison because $0^\circ = 360^\circ$ on a circle. E.g. $\beta = 45^\circ$, $\alpha = 10^\circ$ and $\delta = 90^\circ$; $(\alpha + \delta) = 100^\circ$, $(\alpha - \delta) = 280^\circ$; β is less than $(\alpha + \delta)$ but not greater than $(\alpha - \delta)$. Algorithm 2.3 solves this problem.

Algorithm 2.3 Angle in the Region

INPUT β is the angle to check for
α is the centre of region
δ is the distance from α

inRegion = *true*;

$x = (\alpha + \delta)$;
$y = (\alpha - \delta)$;

//cater for overlapping angles at 0 and 360
IF ($x > 360$)
    IF (($\beta > (x - 360)$) AND ($\beta < y$))
        inRegion = *false*;

IF ($y < 0$)
    IF (($\beta < (y + 360)$) AND ($\beta > x$))
        inRegion = false;

//if there are no overlapping angles
IF (($x < 360$) AND ($y > 0$))
    IF (($\beta > x$) OR ($\beta < y$))
        inRegion = *false*;

RETURN *inRegion*

■ End of Algorithm

Algorithm 2.3 checks if an angle is contained within a region by conducting some routine comparisons. The algorithm returns true if the angle β is between the minimum and maximum boundary and false otherwise.

2.3.3 Volume of Geometric Shapes

2.3.3.1 Sphere

The volume of the sphere ([17]) is given by the formula:

$$V_s = \frac{4}{3}\pi r^3 \tag{2.3.}$$

where r is the radius of the sphere.

2.3.3.2 Cone

The volume of the cone is given by the formula:

$$V_c = \frac{1}{3}\pi a^2 h \tag{2.4.}$$

where a is the cone radius, and h is the height of the cone. When $a = h$ and the cone angle is 90^o, Equation 2.5 becomes

$$V_c = \frac{1}{3}\pi a^3 \tag{2.5.}$$

Calculating the volume of the shaded region in Figure 2.4 is the difference in volume of two right angular cones.

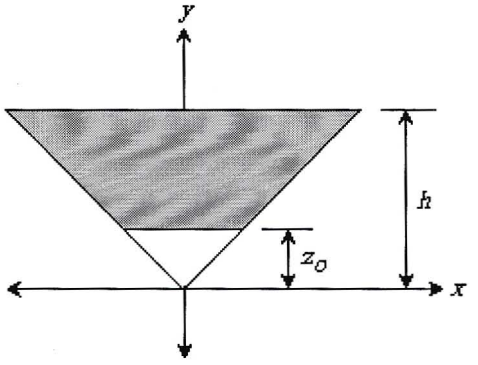


Figure 2.4 Cone

$$V_{sc} = \frac{1}{3}\pi h^3 - \frac{1}{3}\pi z_o^3 \tag{2.6.}$$

2.3.4 Torus

A torus is a mathematical term for donut-shaped region (Figure 2.5). The volume of a torus is obtained by revolving a circle region, R, with radius, r, about the y-axis.

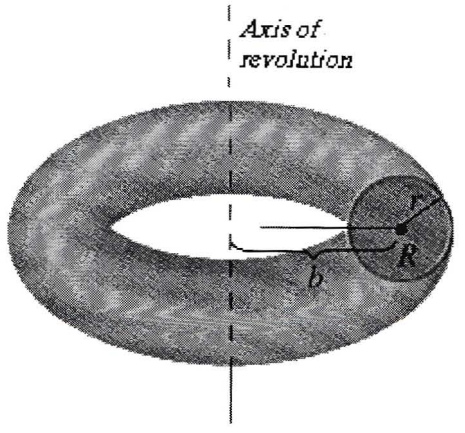


Figure 2.5 Torus

The volume of the torus, [17], is given by

$$V_t = 2\pi^2 r^2 b \tag{2.7.}$$

2.4 OpenGL

OpenGL SuperBible [12] describes the basics of drawing an object to screen. A cube is used to illustrate: perspective, hidden line removal, colours, shading, lights and shadows. These functions are used in this work to show the objects and the interactions of the Two-Body Abrasive Wear process in three-dimension.

OpenGL is strictly defined as "a software interface to graphics hardware." In essence, it is a 3D graphics and modelling library. OpenGL is intended for use with computer hardware that is designed and optimised for the display and manipulation of 3D graphics.

OpenGL is used for a variety of purposes, from CAD engineering and architectural applications to computer-generated dinosaurs in blockbuster movies. The introduction of an industry standard 3D Application Program Interface (API) to a mass-market operating system such as Microsoft Windows has some exciting repercussions. With hardware acceleration and fast PC microprocessors becoming commonplace, 3D graphics is already part of consumer and business applications, not just for games and scientific applications.

2.4.1 How OpenGL Works

OpenGL is procedural rather than a descriptive graphics language. Instead of describing the scene and how it should appear, the programmer actually describes the steps necessary to achieve a certain appearance or effect. These "steps" involve calls to a highly portable API that includes approximately 120 commands and functions. These are used to draw graphics primitives such as points, lines, and polygons in three dimensions. In addition, OpenGL supports lighting, shading, texture mapping, animation, and other special effects.

OpenGL does not include any functions for window management, user interaction, or file input and output. Each host environment (such as Microsoft Windows) has its own

functions for this purpose and is responsible for implementing some means of handing over to OpenGL, the drawing control of a window or bitmap.

2.4.1.1 3D Graphics

"3D computer graphics" are actually two-dimensional images on a flat computer screen that provide an illusion of depth, or a third "dimension." In order to truly see in 3D, the object has to be viewed with both eyes, or each eye must be supplied with separate and unique images of the object. Take a look at Figure 2.6. Each eye receives a two-dimensional image that is much like a temporary photograph on the retina. These two images are slightly different because they are received at two different angles (eyes are spaced apart for this purpose). The brain then combines these slightly different images to produce a single, composite 3D picture, as shown in Figure 2.6.

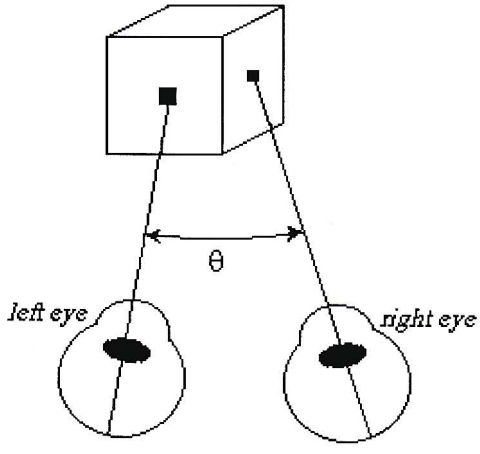


Figure 2.6 How the eyes "see" three dimensions

In Figure 2.6, the angle θ between the images gets smaller as the object goes farther away. This 3D effect can be amplified by increasing the angle between the two images. Stereoscopic viewers and 3D movies capitalize on this effect by placing each eye on a separate lens, or by providing colour-filtered glasses that separate two superimposed images.

2D + Perspective = 3D

The reason the world doesn't become suddenly flat when one eye is covered, is that many of a 3D world's effects are also present in a 2D world. This is just enough to trigger the brain's ability to discern depth. The most obvious cue is that nearby objects appear larger than distant objects. This effect is called perspective. Perspective plus colour changes, textures, lighting, shading, and variations of colour intensities (due to lighting) together add up to the perception of a three-dimensional image.

Perspective alone is enough to lend the appearance of three dimensions. Figure 2.7 presents a simple wireframe cube. Even without colouring or shading, the cube still has the appearance of a three-dimensional object. Stare at the cube for long enough, and the front and back of the cube will switch places.

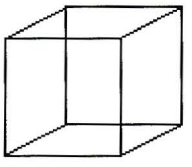


Figure 2.7 Wire frame cube with equal front and back faces.

2.4.1.2 Hidden Line Removal

Figure 2.7 contains just enough information to lend the appearance of three dimensions, but not enough to distinguish the front of the cube from the back. The back is obscured by the front. If the cube in Figure 2.7 were solid, the corners at the back of the cube wouldn't be seen, and thus no confusion with the corners in the front of the cube. Even if the cube were made of wire, parts of the wires in front would obscure parts of the wires at the back. To simulate this in a two-dimensional drawing, lines that would be obscured by surfaces in front of them must be removed. This is called *hidden line removal* and it has been done to the cube in Figure 2.8;

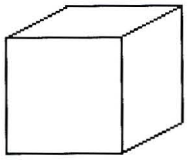


Figure 2.8 The cube after hidden lines are removed

2.4.1.3 Colours and Shading

Figure 2.8 still doesn't look much like a real-world object. The faces of the cube are exactly the same colour as the background, and all that is seen are the front edges of the object. A real cube would have some colour and/or texture. In a wooden cube, for example, the colour and grain of the wood would show. On a computer (or on paper), if only the cube was coloured and drawn in two dimensions, it would be similar to Figure 2.9.

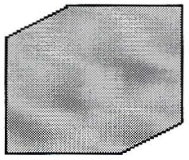


Figure 2.9 The cube with colour, but no shading

The object appears two-dimensional again, and unless the edges are drawn in a different colour, there is no perception of three dimensions at all. In order to regain the perspective of a solid object (without drawing the edges a different colour), either make each of the three visible sides a different colour, or make them the same colour with different shading to produce the illusion of lighting. In Figure 2.10, the faces of the cube all have a different colour or shade.

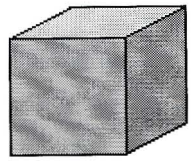


Figure 2.10 The cube with its visible faces in three different shades

Lights and Shadows

One last element that must not be neglected is lighting. Lighting has two important effects on objects viewed in three dimensions. First, it causes a surface of a uniform colour to appear shaded when viewed or illuminated from an angle. Second, objects that do not transmit light (most solid objects) cast a shadow when they obstruct the path of a ray of light. See Figure 2.11.

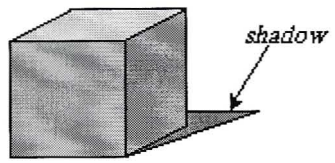


Figure 2.11 A solid cube illuminated by a single light

Two sources of light can influence the three-dimensional objects. *Ambient* light, which is undirected light, is simply a uniform illumination that can cause shading effects on solid objects. Ambient light causes distant edges to appear dimmer. Another type of light is from a light source, called a *lamp*. Lamps can be used to change the shading of solid objects and for shadow effects.

2.4.2 Triangle Strips

Many surfaces and shapes are well represented by a set of connected triangles. Drawing a strip of connected triangles with the Triangle Strip primitive can save a lot of time. Figure 2.12 shows the progression of a strip of three triangles specified by a set of five vertices

numbered V0 through V4. Here the vertices are not necessarily traversed in the same order they were specified. The reason for this is to preserve the winding (clockwise and counter clockwise) of each triangle.

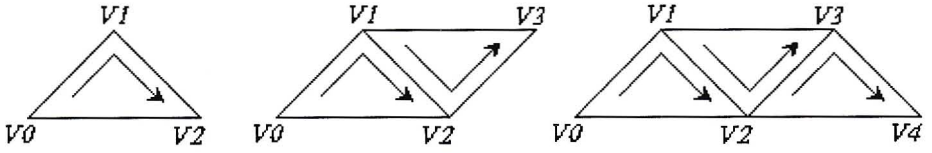


Figure 2.12 The progression of a Triangle Strip

There are two advantages to using a strip of triangles instead of just specifying each triangle separately. Firstly, after specifying the first three vertices for the initial triangle, a single point only needs to be specified for each additional triangle. This saves a lot of time (as well as data space) when many triangles have to be drawn. The second advantage is that it's a good idea to compose an object or surface out of triangles rather than some of the other primitives (three points define a plane).

The HMDS is a 2D array of height values; the triangle strip is used to generate a surface view of the HMDS. Algorithm 2.4 creates a triangle strip for a rectangular surface.

Algorithm 2.4 Creating Triangle Strip for the HMDS

INPUT *surf3Dz* and *surf3Dx* are the surface dimensions

surf3Dx = *surf3Dx**2 –2

FOR *z* FROM 0 TO *surf3Dz*

FOR *x* FROM 0 TO *surf3Dx*

IF ((*z* mod 2) not equal to *0*)

tempX = *surf3Dx* - *x*;

ELSE

tempX = *x*;

tx = *tempX*/2;
tz = *z* + (*tempX* mod 2);

setpoint(tx,tz) //a function to record the points

END for *surf3Dx*

END for *surf3Dz*

■ End of Algorithm

Algorithm 2.4 takes the dimensions of the surface as input, and it output the list of points for the triangle strip. To demonstrate this algorithm consider an array of size *4x4*, this array consist of 16 vertices. The result of creating a triangle strip (by using Algorithm 2.4) for the array is shown in Figure 2.13(a).

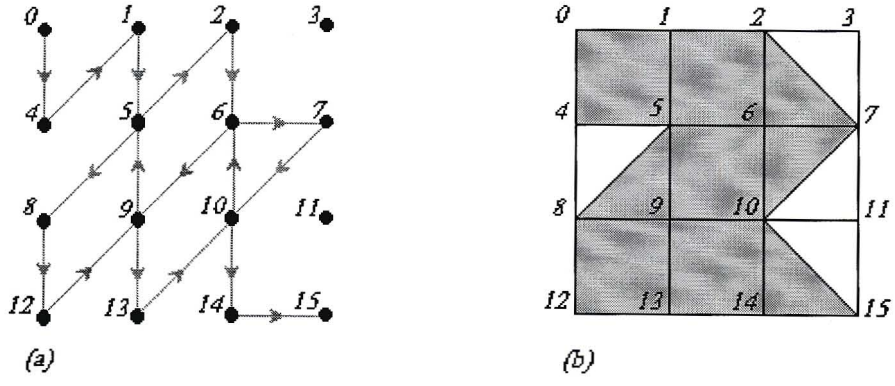


Figure 2.13 Triangle Strip (a) The strip order (b) Area covered by the strip.

The triangle strip Figure 2.13(a) is *0, 4, 1, 5, 2, 6, 7, 10, 6, 9, 5, 8, 12, 9, 13, 10, 14,* and *15*. Algorithm 2.4 is not perfect, some edge vertices are not considered in the triangle strip (e.g. points 3 & 11). This is not a major problem if the arrays are greater than *100x100* in size. Figure 2.13(b) illustrates the area of the surface that is drawn by the triangle strip.

Summary

The mathematical concepts and computational algorithms relevant to this study have been discussed. These concepts and algorithms provide the building blocks for the forthcoming simulation. All methods outlined were chosen for their accuracy and speed.

In the next section we get to the nuts and bolts of the TBAW process.

Chapter 3 The Two-Body Abrasive Wear (TBAW) Process

3.1 Introduction

The movement of one solid surface over another is fundamentally important to the functioning of many mechanisms, both artificial and natural. Whenever surfaces move over each other, damage to one or both surfaces, generally involving progressive loss of material (*wear*) will occur. In most cases, wear is detrimental, leading to increased clearances between moving components, unwanted freedom of movement and loss of precision, often vibration, increased mechanical loading and yet more rapid wear, and sometimes fatigue failure. As in the case of friction, though, high wear rates are sometimes desirable. Grinding and polishing, for example, employ wear processes to remove material rapidly and in a controlled manner, and a small amount of wear is often anticipated and even welcomed during the 'running-in' process in some kinds of machinery.

"A simple model for abrasive wear of ductile material involves the removal of material by plastic deformation. An abrasive particle, idealized as a cone, is dragged across the surface of a ductile material, which flows under the indentation pressure. It forms a groove in the material, and wear is assumed to occur by the removal of some proportion of material, which is displaced by the particle from the groove. The normal load carried by the particle is supported by plastic flow beneath the particle, which causes a pressure to act over the area of contact between the particle and the surface." Hutchings [2]

3.1.1 History

Modelling of wear processes allows the most important factors influencing wear to be identified. In the classical abrasive wear model presented by *Rabinowicz [20]*, hardness is the single most important property that determines the wear rate of the wearing material.

While this model is simple and provides a qualitative theoretical basis, extensive studies have shown that other properties can significantly affect abrasive wear. The predicted wear coefficient (see Section 3.6.3) is at least an order of magnitude larger than experimentally observed values and it cannot predict the effect of grit size on wear and friction coefficients. An abrasion theory considering the material removal has been developed by *Zum Gahr* [21, 22, 23] and has shown good agreement with experimental observations. Most of the existing models mainly deal with single particle contact conditions. *Jacobson et al* [24, 25] applied statistical approaches to simulate multiple grooving processes during two-body wear. In their simulation, the simple classical abrasive wear model was used, pure cutting wear mechanism being assumed. *J. Jiang et al* [8] investigated a model for two-body abrasion of metals under multiple contact conditions. Particle height distribution, wear mode transitions and material removal fraction under various conditions are included in this model.

3.1.2 Terminology

In keeping with the literature the tool grits are referred to as abrasive particles or abrasives.

The "Two-Body" referred to in the title are the **tool** (Figure 3.1) and the **workpiece** (Figure 3.2).

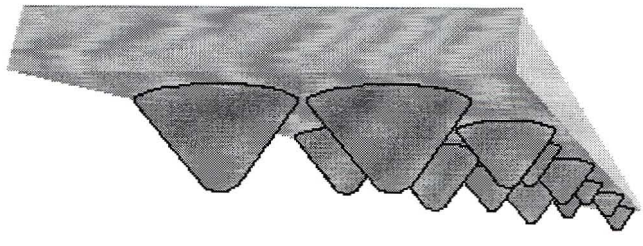


Figure 3.1 Tool

The tool consists of multiple **abrasive particles** (Figure 3.1).

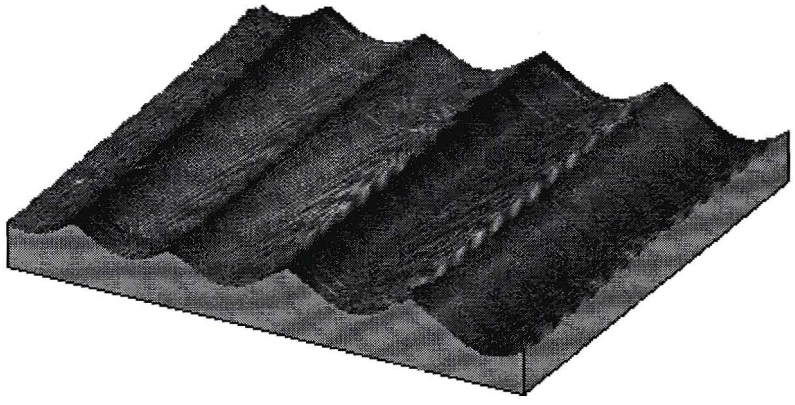


Figure 3.2 Workpiece

The workpiece is a ductile material that exhibits grooves that have been cut by the tool. (Figure 3.2)

3.1.3 Overview

In the TBAW process abrasive particles are randomly distributed on the tool. The abrasive particle's size follows a normal distribution. As a result, under a given load (F), only a fraction of the abrasives on the tool will contact the workpiece surface.

J Jiang et. al. [8] describes the TBAW process as follows. The abrasives penetrate into the workpiece surface at varying depths. Since the diameter size distribution and shape of the abrasives are known, the relationships between the applied force (F), the load-carrying area (A_c) and the relative penetration depth (p) for the abrasive on the workpiece surface can be found. By balancing F and A_c for all the contacting abrasives at a relative approach between the workpiece and tool, the wear volume due to relative movement between the workpiece and tool can be calculated.

3.2 Definitions

3.2.1 Load-Carrying Area

The load-carrying area is the contact area between two bodies that is perpendicular to the applied force. The Force Equilibrium Equation (Equation 3.1) defines workpiece hardness as the ratio of the force to the load-carrying area. From *Jacobson [1]*

$$F = HA_c \tag{3.1.}$$

F is the applied normal force on the tool. H is the workpiece hardness, Section 3.3.1. A_c is the load-carrying area between the tool and the workpiece.

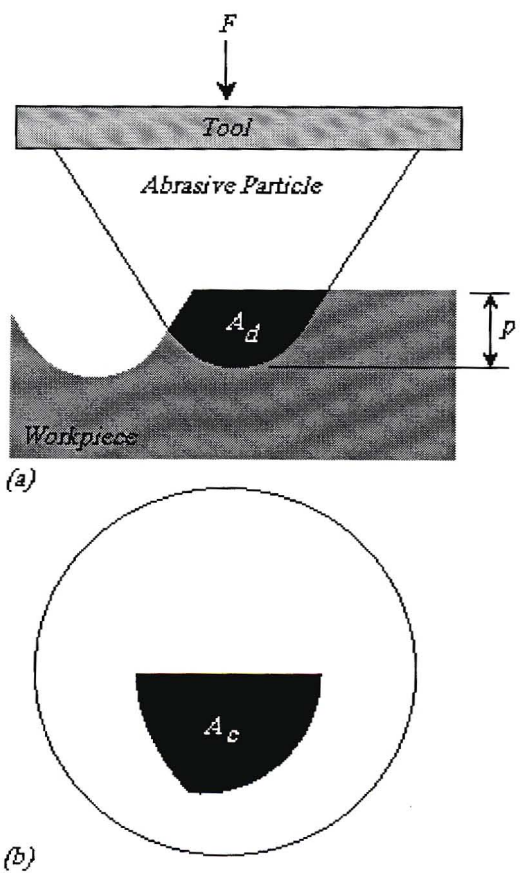


Figure 3.3 Schematic representation of an abrasive penetrating a workpiece [1] (a)Cross-section area of the penetration depth, p. (b) Top view of the load-carrying area, A_c.

Figure 3.3 illustrates a single abrasive tool penetrating the workpiece. Figure 3.3(b) shows A_c between the abrasive and workpiece (assuming the workpiece material is worn off from behind the abrasive).

3.2.2 Displaced Material Volume

The volume of material displaced is of fundamental importance in the wear process. All variables in the TBAW process, directly or indirectly, affect the volume of material displaced.

As the tool slides across the workpiece, material is displaced from the grooves made by the abrasives. Figure 3.3(a) represents the cross-sectional area of the displaced material volume, A_d, this is also known as the cross-sectional area of the wear groove.

3.2.3 Penetration Depth

The penetration depth (p) is the total depth an abrasive is embedded in a workpiece. The relationship between A_c and p is demonstrated in Figure 3.4:

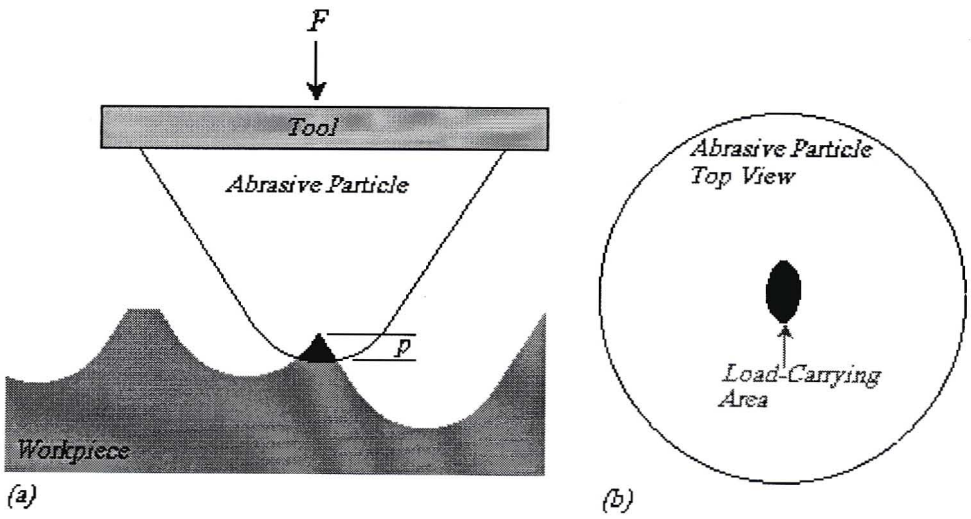


Figure 3.4 Small penetration depth (a) Cross-sectional area of displaced material (b) Top view of the Load-Carrying Area.

From Figure 3.4 a small p between the abrasive and workpiece is shown. The A_c (Figure 3.4(b)) is relatively small. Using the same workpiece profile in Figure 3.4 and increasing p, results in Figure 3.5.

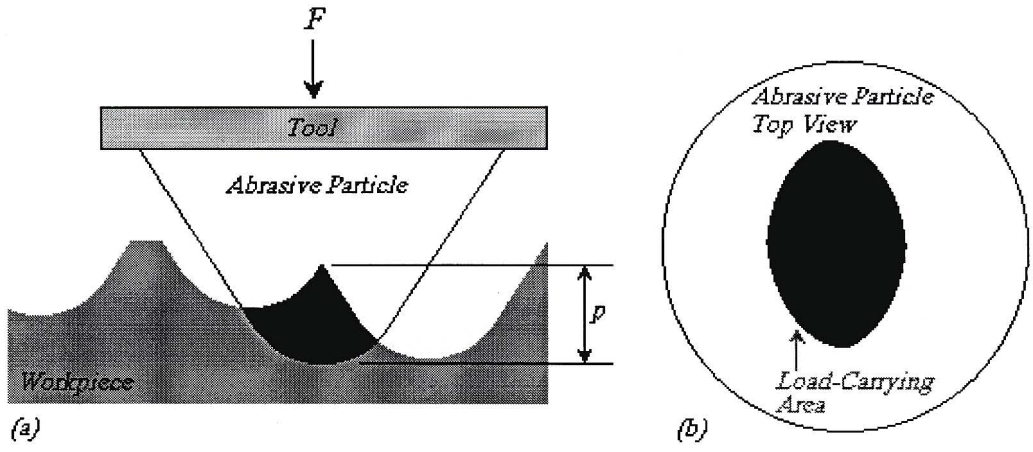


Figure 3.5 Large penetration depth (a) Cross-sectional area of displaced material (b) Top view of the A_c

From Figure 3.4/5 an increase in p results in an increase in A_c (with the same workpiece profile).

3.2.4 Multiple Contacting Abrasives

The tool consists of multiple abrasive particles, each an independent shape and size. Larger tool abrasives have a greater chance of contacting the workpiece, and small abrasives may pass over the workpiece with out any contact. The applied force *(F)* on the tool is distributed amongst the abrasives that penetrate the workpiece at varying depths (Figure 3.6). Contact between the abrasives and the tool occurs only at the front of the abrasives, due to the moving tool.

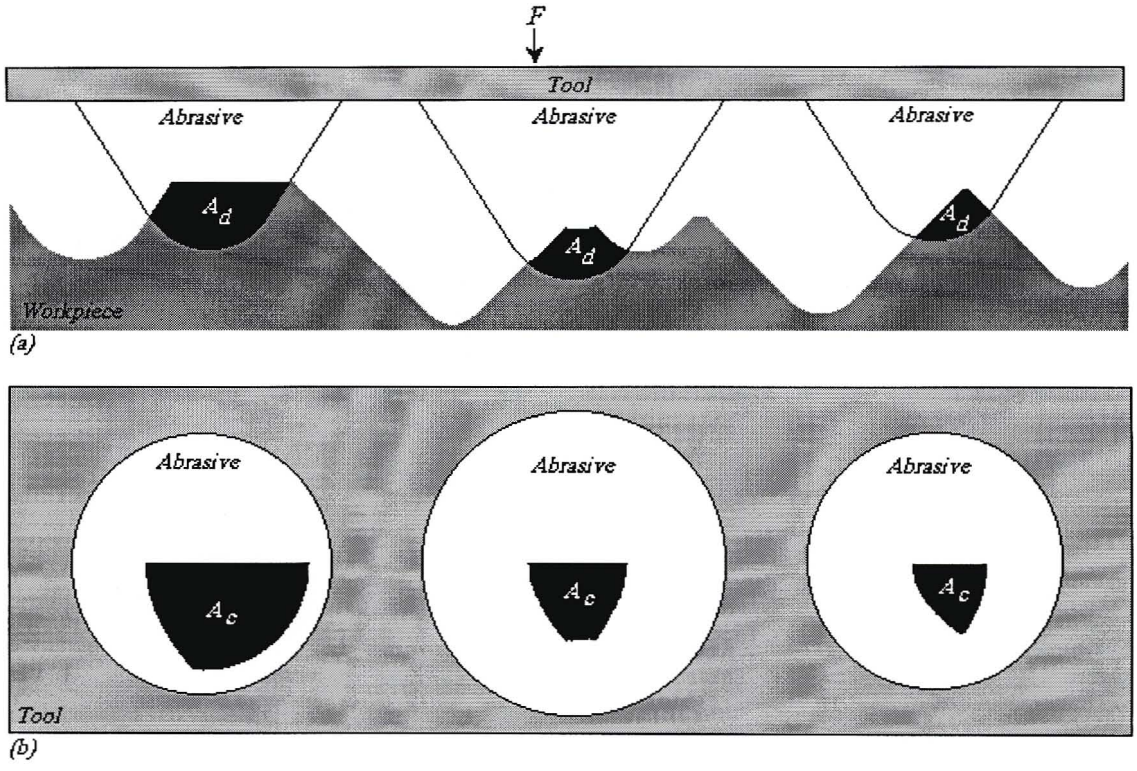


Figure 3.6 Multiple contacts. (a) Cross-section area of displaced material. (b) Top view of the A_c

Figure 3.6(a) represents the cross-sectional area of displaced material, which is perpendicular to the grooving direction. Figure 3.6(b) is the top view of the tool that shows each abrasive and the corresponding A_c.

3.3 Abrasive Properties: Hardness, Shape and Size

3.3.1 Hardness

The hardness of a material is defined as $\frac{F}{A_c}$ (Applied Force over Load-Carrying Area).

"The hardness of the abrasives involved in abrasion influences the wear rate: abrasives with lower hardness than that of the workpiece cause much less wear than the harder abrasives. For abrasives significantly harder than the workpiece, the exact value of their hardness matters much less.

The wear rate becomes much more sensitive to the ratio of abrasive hardness H_a to the workpiece surface hardness H_s when H_a /H_s is less than 1. The contact pressure depends mostly on the indentation hardness of the workpiece, and depends little on the detailed shape of the abrasive. Plastic indentation of the workpiece will occur as the normal load on the tool is increased only if the abrasives can sustain this contact pressure without deforming or fracturing.

It is observed experimentally that abrasive grit particles of any shape will cause plastic scratching only if H_a /H_s > 1.2. Abrasion under conditions where H_a /H_s< 1.2 is sometimes termed soft abrasion, in contrast to hard abrasion when H_a /H_s > 1.2." Hutchings [2]

Greater wear rates result when the abrasive particles have more than about 1.2 times the hardness of the workpiece. The H_a in this TBAW was assumed to be much greater than H_s. All contacts were thus plastic in terms of the response of the workpiece to the abrasive.

3.3.2 Shape

The second important influence on the wear rate is the abrasive particle shape. *J. Jiang et al [8]* investigated the effect on the wear rate by varying abrasive shape (rounded to angular). Wear rates depend strongly on the shapes of abrasives, with angular abrasives causing greater wear than rounded abrasives. (Only for the case of H_a /H_s > 1.2).

3.3.3 Size

The sizes of abrasive particles cover a wide range. The size of the abrasive particle has an indirect influence on the wear rate. From a mathematical perspective, larger abrasive particles decrease the number of particles on the tool (density). Fewer abrasives would be in contact with the workpiece (assume the same applied Force), therefore a greater

penetration depth (removing a larger volume of material from the workpiece) and increasing the wear rate.

The effect of the abrasive size on the wear rate is known as the *grit size effect/phenomena* and has been studied by a number of investigators. [1,8, 23,26,27]

3.4 Contact Theory

3.4.1 Contact

When two nominally plane and parallel surfaces are brought gently together, contact will initially occur at only a few points. As the normal load is increased, the surfaces move closer together and a larger number of the higher areas *(Abrasive Particles)* on the surface come into contact. Since these abrasives provide the only points at which the surfaces touch, they are responsible for supporting the normal load on the surface and for generating any frictional forces which act between them, *Hutchings [2]*. An understanding of the way in which the abrasives interact under varying loads is therefore essential to any study of wear.

3.4.1.1 Plastic Deformation of the Plane

Plastic deformation is demonstrated by sphere, H_a, penetrating a plane, H_s, under a normal load (Figure 3.7).

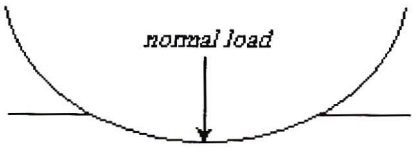


Figure 3.7 Sphere indenter under a load

As illustrated in Figure 3.7, there were no deformations of the sphere under the normal load, only plastic deformation of the plane. The exact shape of the sphere was imprinted in the plane. Material from the plane is either removed or displaced to the side of the

abrasive groove. Fully plastic deformation only occurs when the difference in hardness between the two surfaces is large (i.e. $H_a / H_s > 1.2$). *Johnson [4]*

3.4.2 Workpiece Interaction with Sliding Abrasives

Only a portion of the volume of a wear groove formed on a workpiece surface is removed as wear debris and the remainder of the groove volume is plastically displaced to the sides of the groove. In this work microploughing and microcutting are simulated.

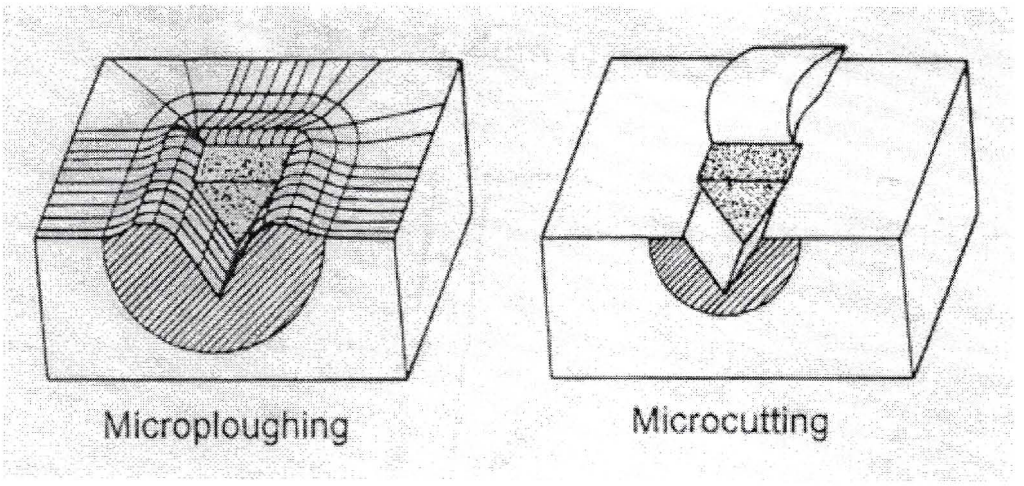


Figure 3.8 Schematic representation of different interactions between sliding abrasive particles and the workpiece surface.

"A general TBAW model describes abrasive wear by distinguishing two types of interactions between abrasive particles and a wearing material, namely microploughing and microcutting (Figure 3.8). In the ideal case, microploughing due to a single pass of one abrasive particle does not result in any detachment of material from the workpiece surface. A prow is formed ahead of the abrading particle and material is continuously displaced sideways to form ridges adjacent to the groove produced. Volume loss can, however, occur owing to the action of many abrasives or the repeated action of a single abrasive. Pure microcutting results in a volume loss by chips equal to the volume of the wear grooves. Microploughing and microcutting are the dominant processes on ductile material." Zum Gahr [9]

3.4.3 Measuring Microploughing and Microcutting

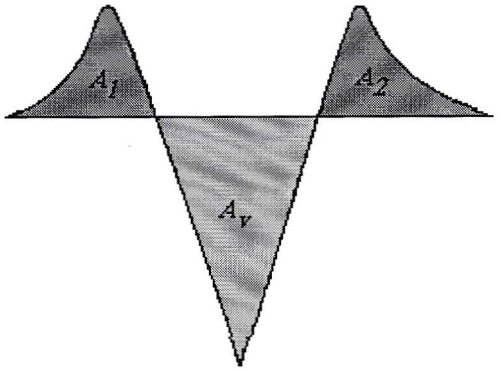


Figure 3.9 Cross-Section though a wear groove

Zum Gahr [9] measures microploughing and microcutting as follows. The ratio of material volume removed as wear debris ($A_1 + A_2$) to the volume of the wear groove (A_v) produced can be described by the f_{ab} value (Figure 3.9), which is defined as

$$f_{ab} = \frac{A_V - (A_1 + A_2)}{A_V} \tag{3.2.}$$

where A_V is the cross-section area of the wear groove and $A_1 + A_2$ represents the amount of material which is pushed to the groove sides by plastic deformation. The f_{ab} values become equal to unity for ideal microcutting and equal to zero for ideal microploughing.

3.5 Tool

3.5.1 Tool Creation

The tool consists of two component types; a solid base and abrasive particles. Each of the abrasives is bonded to the solid base without intersecting each other. The abrasives have a specific size and shape distribution.

Depending on the technique and material used for obtaining the abrasives, the shape of abrasives created may be manipulated. Each abrasive is randomly positioned (*uniform distribution*) on the tool base.

3.5.2 Tool Shape

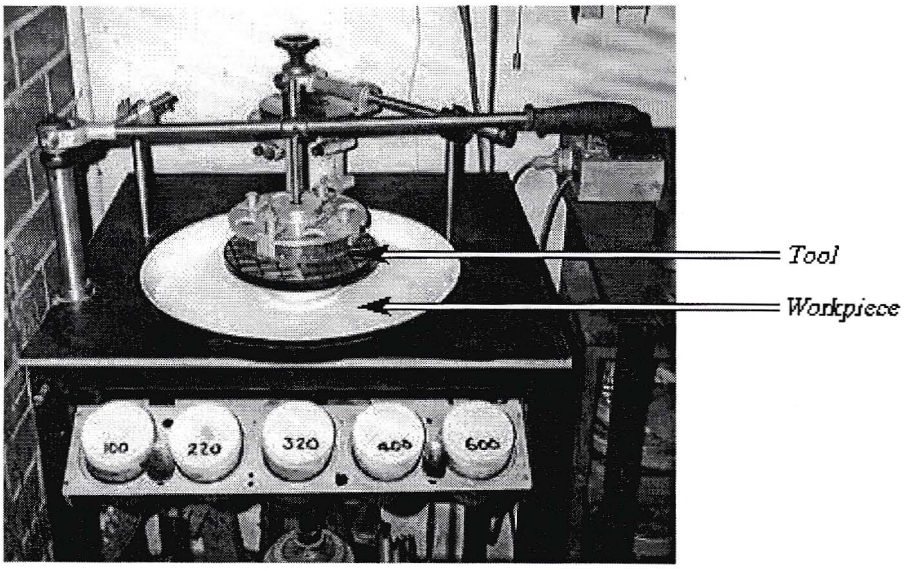


Figure 3.10 An experiment of the TBAW process [28]

Figure 3.10 is a photograph of the polishing process. The tool spins and slides over the entire workpiece area. Due to the limitations of computing power only a small portion of the tool is examined in this research.

3.5.3 Spinning Disc Path

The path of the spinning disc sliding over the workpiece is shown in Figure 3.11. There are two independent movements related to the spinning disc, the rotation (Figure 3.11(a)) and the movement backwards and forwards (Figure 3.11(b)). These two motions combined allow each abrasive on the tool to traverse a large workpiece area.

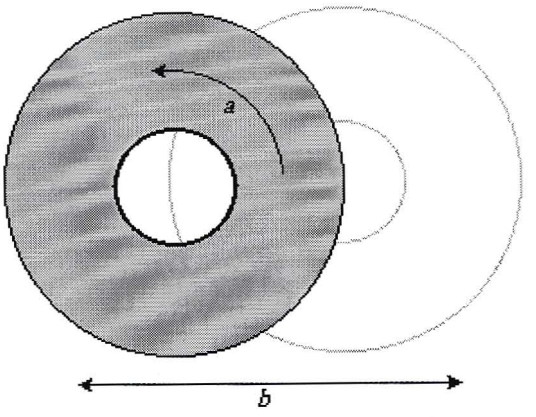


Figure 3.11 Spinning moving disc

3.6 Test Methodology

3.6.1 Surface Roughness

Surface roughness is the deviation of the actual surface topography from an ideal atomically smooth and planer surface. A measure of the surface roughness is the mean deviation from the centre line average.

The graph of a surface profile generated by a stylus or optical profilmeter[2] contains most of the information needed to describe the topography of the surface along a single direction. The profile graph itself, however, does not provide a sufficiently simple and readily interpreted means of describing surface roughness; several quantities derived from the profile, which are often automatically computed by the profilmeter instrument, are used for this purpose.

The most commonly quoted measure of surface roughness is the *average roughness* R_a (centre line average or arithmetic average.). R_a is defined as the arithmetic mean deviation of the surface height from the mean line through the profile. The mean line is defined so that equal areas of the profile lie above and below it (Figure 3.12).

[2] A fine stylus is dragged smoothly and steadily across the surface under examination. As the stylus travels over the surface it rises and falls, and its vertical displacement is recorded and digitised for computer processing. [2]

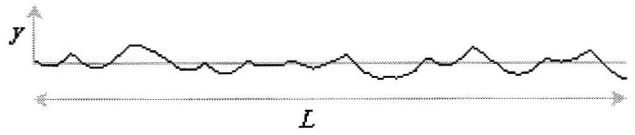


Figure 3.12 A surface profile is a graph of surface height, y, relative to a mean line, plotted against distance. The overall length of the profile under examination is L

Formally, the average roughness R_a is defined by

$$R_a = \frac{1}{L}\int_0^L |\, y(x) \,|\, dx \tag{3.3.}$$

Where *y(x)* is the height of the surface above the mean line at a distance x from the origin, and L is the overall length of the profile under examination. [2]

3.6.1.1 Three-Dimensional Extension

Extending R_a to a surface with relief entails a number of changes. The profile surface contains the surface height value at each point. R_a is defined as the arithmetic mean deviation of the surface height from the mean plane through the surface profile. The mean plane is defined so that equal volumes of the profile lie above and below it.

The average roughness R_a is defined by

$$R_a = \frac{1}{bc}\int_0^b \int_0^c |\, y(x,z) \,|\, dzdx \tag{3.4.}$$

y(x, z) is the height of the surface above the mean plane at position (*x, z*), *bc* is the normal area of the plane.

3.6.2 Wear Rate

Wear rate, Q, is the volume of material removed, V, per unit sliding distance, s. [2]

$$Q = \frac{V}{s} \tag{3.5.}$$

Q unit of measurement is $mm^3\ m^{-1}$.

Dimensionless wear rate is define as $\frac{Q}{A_c}$, where A_c is the normal contact area between the tool and workpiece, see Section 3.2.1. Dimensionless wear rate makes it easier to compare results with other authors.

3.6.3 Wear Coefficient

$$Q = \frac{KF}{H} \tag{3.6.}$$

Equation 3.6 relates the volume worn per unit sliding distance, Q, to the normal load, F, and the hardness of the softer surface (workpiece), H. This is often called the ***Archard Wear Equation***. The constant K, usually termed the *wear coefficient*, is dimensionless.

$$K = \frac{QH}{F} \tag{3.7.}$$

The dimensionless wear coefficient K is of fundamental importance, and provides a valuable means of comparing the severity of wear process in different systems. However, for engineering applications the quantity K/H is often more useful.

$$k = \frac{K}{H} = \frac{Q}{F} \tag{3.8.}$$

The symbol k is called the *dimensional wear coefficient.* k is usually quoted in units of mm^3 $(Nm)^{-1}$, and represents the volume of material removed by wear (in mm^3) per unit distance slid (in meters), per unit normal load on the contact (in Newtons). The measure of wear provided by k is particularly helpful for comparing wear rates in different classes of material. In some materials, elastomers for example, there are basic problems with the use of the dimensionless coefficient K since the plastic indentation hardness, H, cannot be defined, *Hutchings [2].*

3.7 Previous Simulation Attempts

3.7.1 Jacobson et al

A numerical model was developed to simulate two-body abrasion and investigated by *Jacobson et al [1].* Their model consists of abrasive particles (*conical with hemispherical tips*) and one piece of abraded material (*workpiece*). Successive conical tips grooving in the workpiece simulate abrasion. The profile and average level of the workpiece surface was displayed in the form of a two-dimensional cross-section perpendicular to the grooving direction. A new profile and average level were calculated after each single abrasive (conical) tip has passed the cross-section.

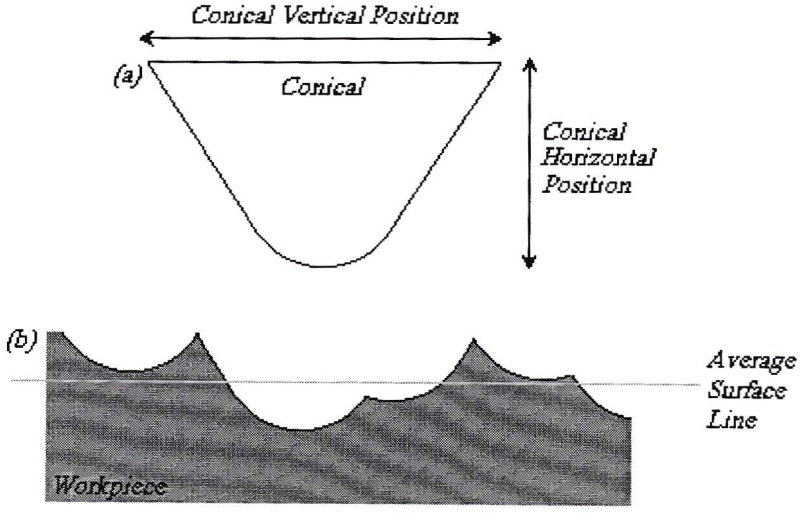


Figure 3.13 Schematic representation of two-body model (a) Conical and its position (b) Workpiece profile.

Before each grooving event a random generator was used to select conical parameters:

1. Horizontal position.
2. Vertical position, a distance from the average surface level of the workpiece.

As in Figure 3.13.

The conical penetration depth, p, was influenced by the initial surface profile of the workpiece and the position of the grooving conical. Because of the distribution of conical size, a large part of all conical tips pass above the workpiece surface, as in a practical case.

For each grooving event the *area of groove cross-section*, A_d, and the projected *Load-carrying area*, A_c, were calculated and stored (Figure 3.13).The surface topography, wear rate, wear coefficient and specific grooving energy were recorded.

The model was restricted to pure microcutting without material displacement to the sides of the groove. All considerations were geometrical and material parameters were not included, except for the homogeneous workpiece hardness.

3.7.2 Jiang et al

This is a general model for wear rate and specific wear energy of materials in the two-body abrasive wear under multiple contact conditions, investigated by *Jiang et al* [8]. This model was further evaluated under various conditions by assuming conical particles with round tips.

Refer to Figure 3.14. Given the height distribution and shape of the abrasives particles and the relationship between the applied load on a single particle. The relative penetration depth of the particle to the workpiece surface is found, by balancing the applied normal

load and the load carried by all the contacting particles at a certain position of the workpiece surface, h_z.

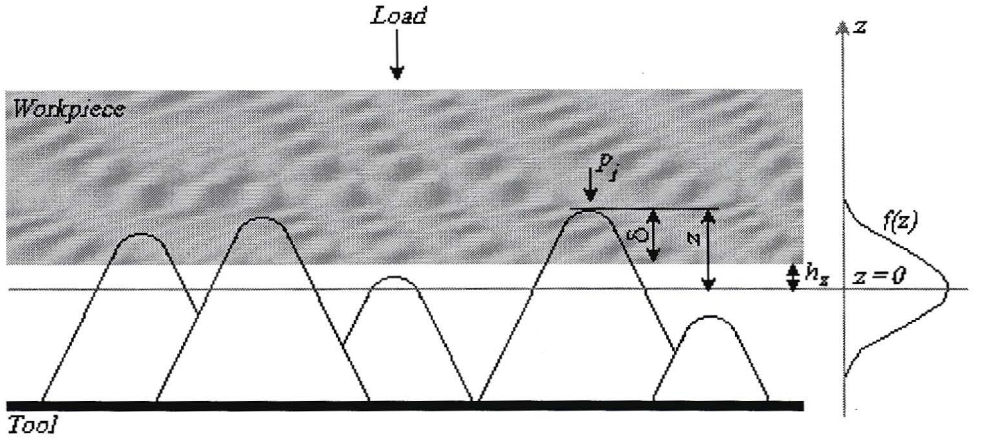


Figure 3.14 Multiple contacting abrasive particles (from Jiang et al)

Using the coordinate system shown in Figure 3.14, with a normalised height distribution of particles on the abrasive paper, $f(z)$, the number of particles, whose heights fall within the height region, $[z, z + dz]$, is equal to $Nf(z)dz$[3]. Under a given contact pressure, p, the hard abrasive particles will penetrate into the workpiece surface until the load carried by all the particles, P_i, contacting the workpiece surface balances the total applied load. According to this load balance condition, the position of the workpiece surface, h_z, can be determined, from the equation:

$$p = \int_{h_z}^{\infty} [Nf(z)]P_i(z - h_z)dz \qquad (3.9.)$$

This model *Jiang et al [8]* was evaluated under various conditions by assuming conical particles with rounded tips. The critical grit size phenomenon (The grit size has little or no influence on the wear rate for small conical tip radius) was predicted. Based on simulation results, Jiang et al show that the presence of a critical grit size mainly results from the rounded feature of the particle tip.

---

[3] Where N is the total number of particles and $f(z)$ is normalized so that $\int f(z)dz = 1$

3.8 Summary

The TBAW process has been defined and discussed. Previous attempts at modelling the process have been outlined. In the next section we outline a new graphical simulation for the TBAW process.

Chapter 4 A New Computer Simulation for the Two-Body Abrasive Wear Process

4.1 Introduction

Numerical and statistical (N&S) models have been applied to the two-body abrasive wear (TBAW) process by a number of authors [1, 8, 9, 18, 20, 26, 27, 29, 30, 31]. These N&S models are exceptionally good, but lack the interactivity and visualisation of a three-dimensional (3D) computer simulation. Owing to the recent developments in computer technology there is still much scope for research in modelling the TBAW process.

This chapter focuses on the conversion of physical TBAW objects to their digital representation. The rules for modelling the events that cause the wearing process are also described.

4.1.1 Surface Representation

The tool and workpiece surfaces of the TBAW process are the only surfaces displayed in the simulation. The TBAW shape of the tool is circular, and therefore the shape of the workpiece surface in contact with the tool is also circular. To simplify matters the circular shaped surface (disc) is rearranged to form a rectangular shape. Figure 4.1 illustrates the geometric representation of the tool.

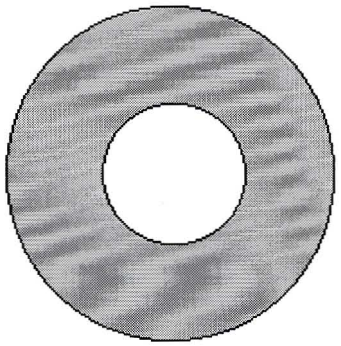


Figure 4.1 Disc without the centre section

The disc is divided at equal intervals, as in Figure 4.2. The number of intervals on the disc is proportional to the object's resolution and accuracy in the simulation.

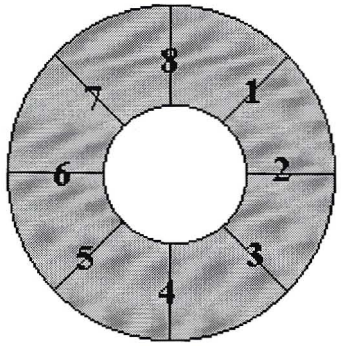


Figure 4.2 Disc divided into eight profiles, at equal intervals apart

At each dividing point the profile is recorded. These eight profiles are then arranged in a rectangular shape (Figure 4.3).

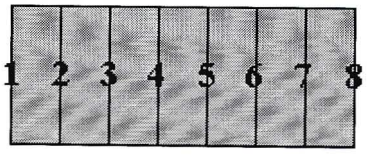


Figure 4.3 Rectangular arrangement of the profiles

The path travelled by a point on the rotating disc is a straight line in the rectangular representation.

4.2 Tool

A digital model of the TBAW tool (Section 3.5) is constructed. The method used to generate each conical is described, with their data structures. For the purpose of this study abrasive particles are represented by conicals. The algorithm used to select, and position each conical is also discussed.

4.2.1 Conical

4.2.1.1 Conical geometry

The conical with a hemispherical tip (Figure 4.4) has been used over the past 20 years for research in this field (TBAW modelling), to represent the abrasive. The conical represents the average shape of abrasives, and reduces the geometric complexity in past N&S models.

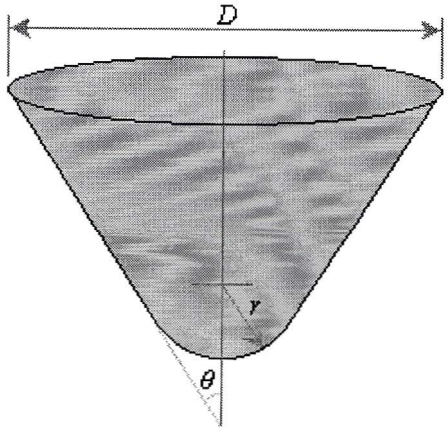


Figure 4.4 Conical with hemispherical tip

The size of a particle is measured by its diameter, D, while its shape is described by the hemispherical tip radius, r, and the cone angle, θ. [1]

4.2.1.2 Conical Data Structure

The *height map data structure,* HMDS (see Section 2.2.3), is used to represent the conical.

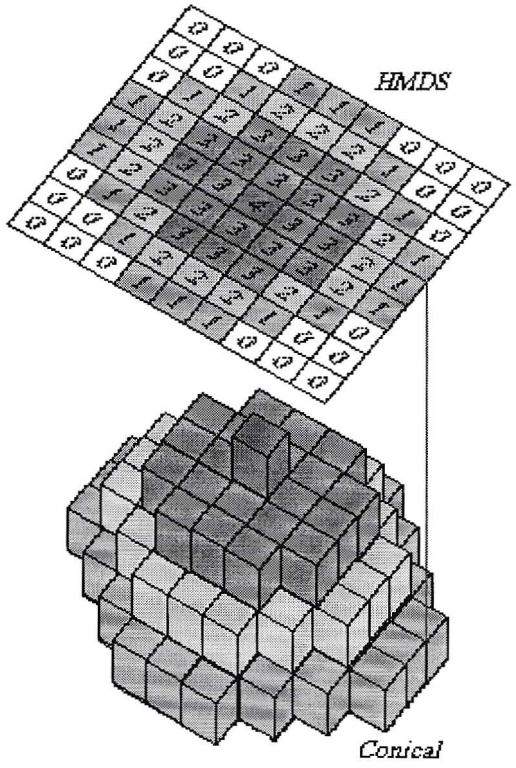


Figure 4.5 HMDS representation of a conical

In Figure 4.5 is a low resolution conical and its HMDS representation is given.

4.2.1.3 Generating the Conical

Using the HMDS to generate the 3D conical shape combines two basic shapes; the cone and sphere. From Figure 4.4 the sphere is at the tip of the conical and the cone forms the rest of the conical shape.

The two basic shape equations are;

Cone: $x^2 + y^2 = z^2$ (4.1.)

Sphere: $x^2 + y^2 + z^2 = r^2$ (4.2.)

The *x, y* and *z* in equations 4.1/4.2 refer to a point in 3D space and *r* is the sphere radius. Solving equations 4.1 and 4.2 for *z* yields

$$z = \pm\frac{r}{\sqrt{2}} \tag{4.3.}$$

Only the positive *z* value is used. Thus $z_0 = \frac{r}{\sqrt{2}}$, is the vertical (z-axis) intersection point for the cone and sphere (Figure 4.6(a)).

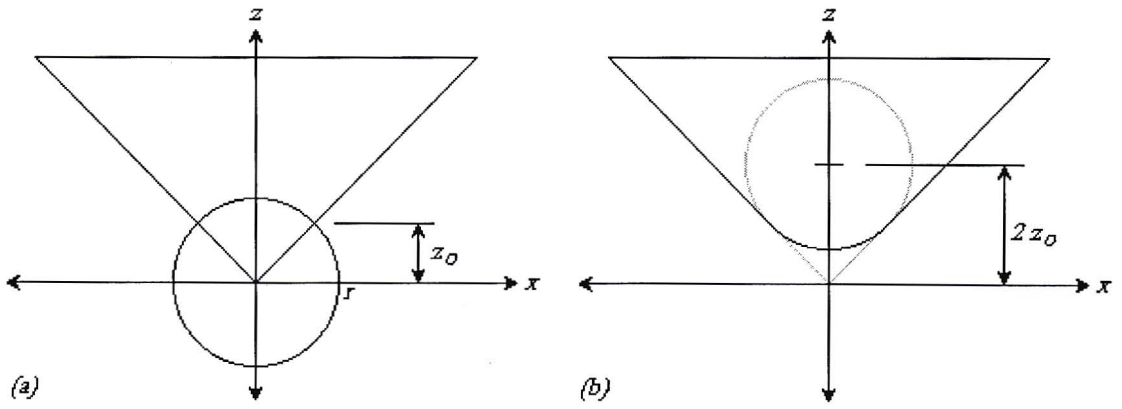


Figure 4.6 Sphere and cone intersection (a) Critical intersection value, z_o (b) Translated sphere

Translating the sphere in the *z-direction* by $2z_o$ creates a smooth conical shape as in Figure 4.6(b).

Algorithm 4.1 Generating the Conical

INPUT *max* is the maximum conical radius

r is the radius of the sphere

conical is the HMDS, with each element assigned to 2**max*

$$z_o = \frac{r}{\sqrt{2}}$$

FOR *x* FROM *0* TO *(2*max)*

FOR *y* FROM *0* TO *(2*max)*

//construct the cone portion of the conical using the cone equation

FOR *z* FROM z_o TO *max*

IF $((x^2 + y^2) \leq z^2)$ AND $(z < conical[x, y])$

$conical[x, y] = z$

//construct the hemisphere portion of the conical using the sphere equation.

FOR *z* FROM -*r* TO 0

IF $((x^2 + y^2 + z^2) \leq r^2)$ AND $((z + 2\,z_o) < conical[x, y])$

$conical[x, y] = z + 2\,z_o$

END FOR *y*

END FOR *x*

RETURN *conical*

■ End of Algorithm

By selecting the *r* and *max* values Algorithm 4.1 generates a smoothly blended digital conical, Figure 4.7.

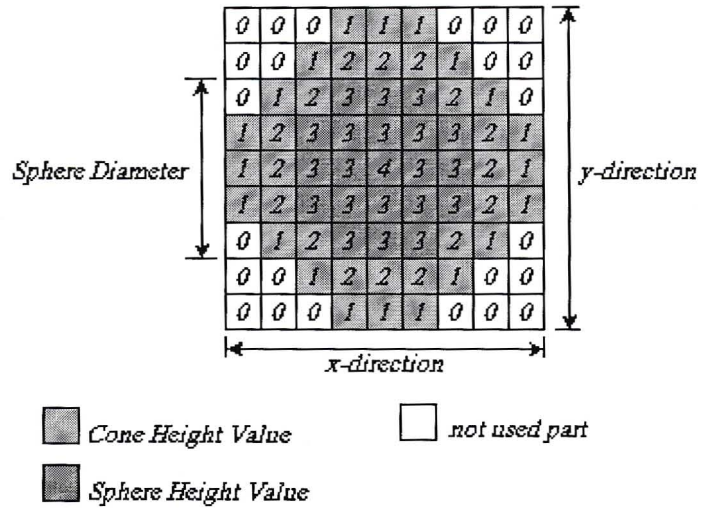


Figure 4.7 HMDS of the Conical, r = 3 and maxRad = 5.

4.2.1.4 Conical Shape

Another method of representing the different abrasive shapes is to include a scale factor. By changing the scale factor of the unit conical (where the scale factor is 1), its cone angle and tip radius also changes. The conical has to resemble both the elongated and flat shapes of abrasives.

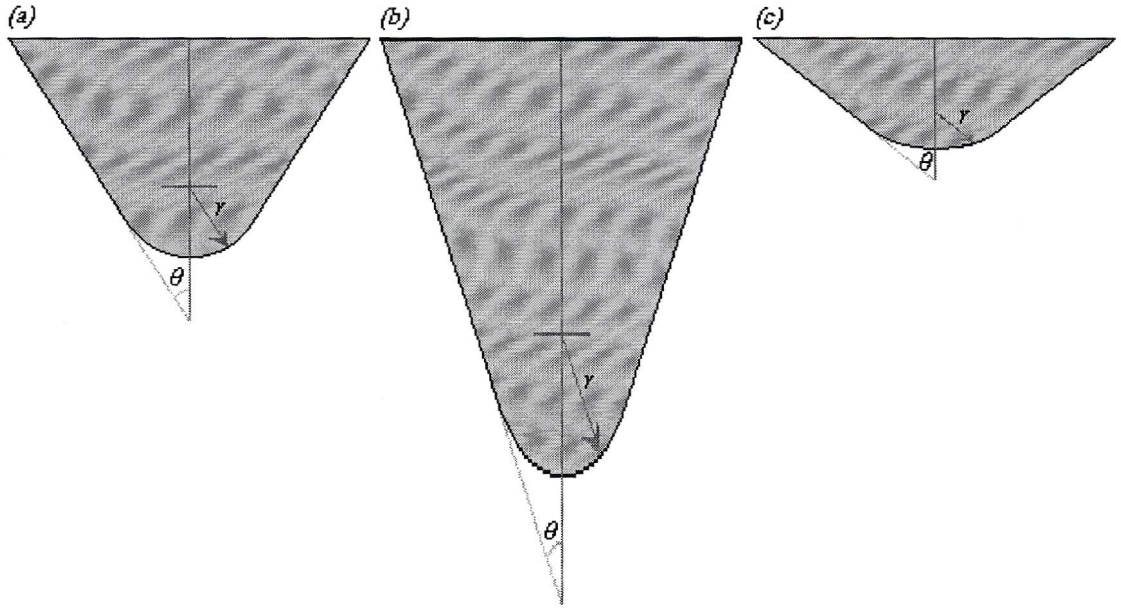


Figure 4.8 Scaled conical (a) Unit conical (b) Scaled in the y-direction by a factor 2 (c) Scaled in the y-direction by a factor 0.5.

Figure 4.8 illustrates three different conical shapes. *(a)*[4] is the unit conical, $\theta = 45^0$ and $r = 10$ *units.* All conicals are derived from the unit conical by specifying a scale factor and diameter. *(b)* is a result of scaling the unit conical in the y-direction by 2, $\theta = 26.5^0$ and $r = 20$ *units. (c)* is obtained by scaling the unit conical in the *y-direction* by 0.5, $\theta = 63^0$ and $r = 5$ *units.* After scaling, the hemispherical tip radius value, r, is not correct; r now represents the average tip radius value.

4.2.2 Tool Base

The conicals are bonded onto the tool base to form the tool. This is an abstract element that keeps all the conicals within an area, and at the same base height. The number of conicals on a tool is proportional to the tool base area *(assuming D is constant).* The conicals do not intersect on the tool base (Figure 4.9).

[4] Sub notation (a), (b) and (c) in this paragraph refer to Figure 4.8.

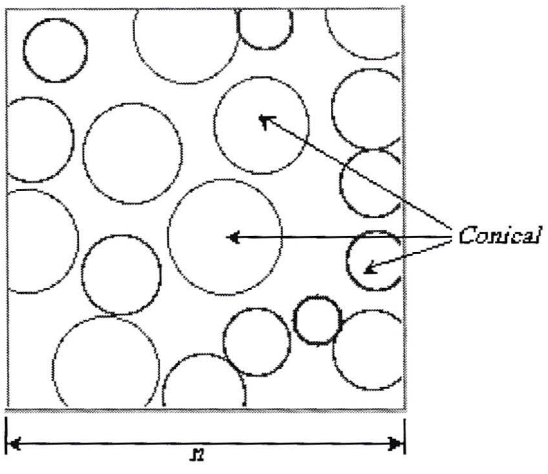


Figure 4.9 Square (n x n) tool base area

If a portion of the conical lies outside the tool base area (Figure 4.9), this portion is not considered as part of the tool.

4.2.3 Tool Construction

A tool is constructed by specifying the following variables: size of tool base area, total volume of conicals, mean conical diameter and standard deviation.

4.2.3.1 Conical Volume

The conical is partially a cone and partly a sphere; therefore calculating the total volume of the conical is the sum of the volume of the partial sphere (Section 2.3.5.1), V_s, and the partial cone (Section 2.3.5.2), V_c. (Figure 4.10)

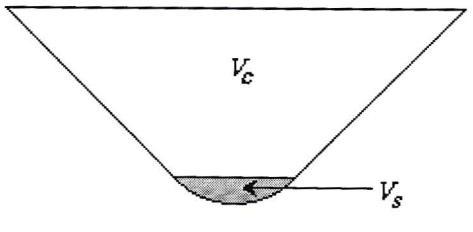


Figure 4.10 Volume of conical

4.2.3.2 Total Conical Volume

This refers to the total volume of conicals on the tool. Consider the TBAW process, the tool abrasive volume is

$$A_{vol} = \frac{A_{mass}}{A_{den}} \tag{4.4.}$$

Where A_{vol} is the total volume of abrasive particle, A_{mass} is the total mass of the abrasives and A_{den} is the molecular density of the abrasive material.

The total conical volume is equal to A_{vol}.

4.2.3.3 Standard Normal PDF

In the TBAW process, the tool does not have identical abrasives. The tool consists of abrasives of a size range (size distribution). This simulation imitates the size distribution by using the *standard normal probability density function* (Equation 4.5).

$$g(x) = e^{-(x-\mu)^2/2\sigma^2} \qquad -\infty < x < \infty \tag{4.5.}$$

With a given mean (μ) and standard deviation (σ), substituting suitable x values produced a *normal distribution.* μ is the measure used to describe the centre of the distribution and σ is the measure used to describe the spread of the distribution. [15]

A characteristic of $g(x)$ is that it always produces a number between 0 and 1. If x is close to μ, $g(x)$ would be close to 1. The rate at which $g(x)$ approaches 0 with x values away from μ depends on σ value, as in Figure 4.11.

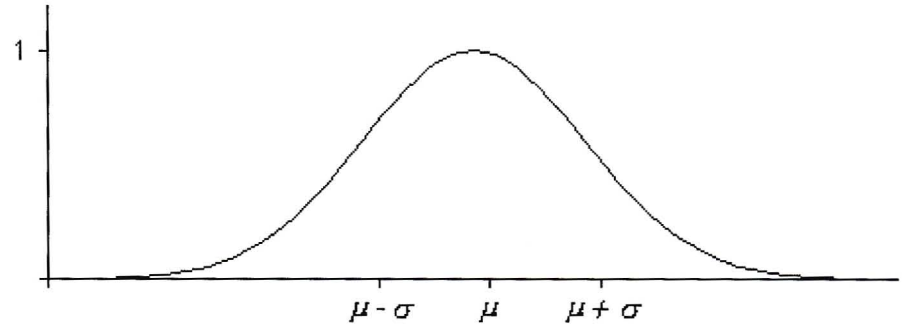


Figure 4.11 The graph of g(x). [15]

4.2.3.4 Selecting the Conical

Each conical is placed at a random position on the tool base, but the conical diameter follows a *normal distribution.*

Algorithm 4.2 Creation of the Tool.

INPUT *Max_Volume*

WHILE (*Max_Volume* > 0)

a = random value between [*0*, *max diameter size*]
b = random value between [*0*, *1*]
g(x) = returns the standard normal pdf for *x*.

(*n* x *n*) = Tool Base Area dimension, *0* < *n*
pos = random position between [(*0*, *0*), (*n*, *n*)]

Intersect(*pos*, *a*) takes the size and position of the conical, returns true if there is an intersection with another conical already on the tool base (Section 2.4.2).

IF ((*g*(*a*) ≥ *b*) AND (NOT Intersect(*pos*, *a*)))

ACCEPT the new conical at position *pos,* and size *a.*

Max_Volume -= Volume(a)

RETURN list of conical with their size and position on the tool.

■ End of Algorithm

Algorithm 4.2 takes the volume of conical required on the tool as input. The random size and position of the conical is chosen, as well as another random value, *b*. If the conical does not intersect another conical on the tool base and its size follows a normal distribution then the conical is either excepted or not. This process is repeated until the tool contains the total volume of conicals.

4.2.3.5 Tool Data Structure

The tool does not have a separate data structure, but a combination of the HMDS (for the unit conical) and a few variables. These variables include the diameter size (*D*), scale and position for each of the conicals on the tool.

4.3 Workpiece

The two variables for the workpiece are the hardness and the initial surface roughness. The hardness value represents a specific type of material (Section 3.3.1). The nature of the initial workpiece surface is a variable in the TBAW process. Different workpiece surface roughness could be selected prior to abrading. The surface roughness is represented as a HMDS.

4.4 Wear

Wear is the most important section in this research. Wear deals with the interaction between the two objects (*Tool and workpiece*). As the tool sides over the workpiece, at each new position the load carrying area, A_c, is found that satisfies the Force Equilibrium Equation (Equation 3.1). The penetration depth is calculated and the workpiece is worn

(new groove formed). Depending on the hardness of the workpiece, material is either ploughed to the sides of the groove and/or removed from the workpiece.

4.4.1 Contact

All contacts between the tool abrasives and workpiece are plastic deformations. This implies negligible wear and extremely high hardness values for abrasives, i.e. $H_a / H_s > 1.2$ (Section 3.3.1).

The contact area is defined as the tool's (*conical*) area that touches the workpiece (Figure 4.12).

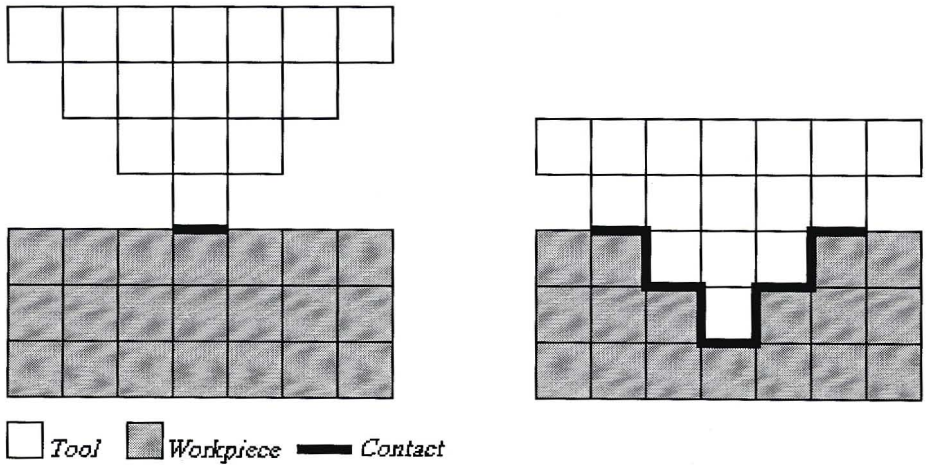


Figure 4.12 Cross-sectional area of the tool and workpiece, the contact represented by the thick black line.

4.4.2 Load-Carrying Area

The load-carrying area, A_c, is the contact area perpendicular to the applied normal force.

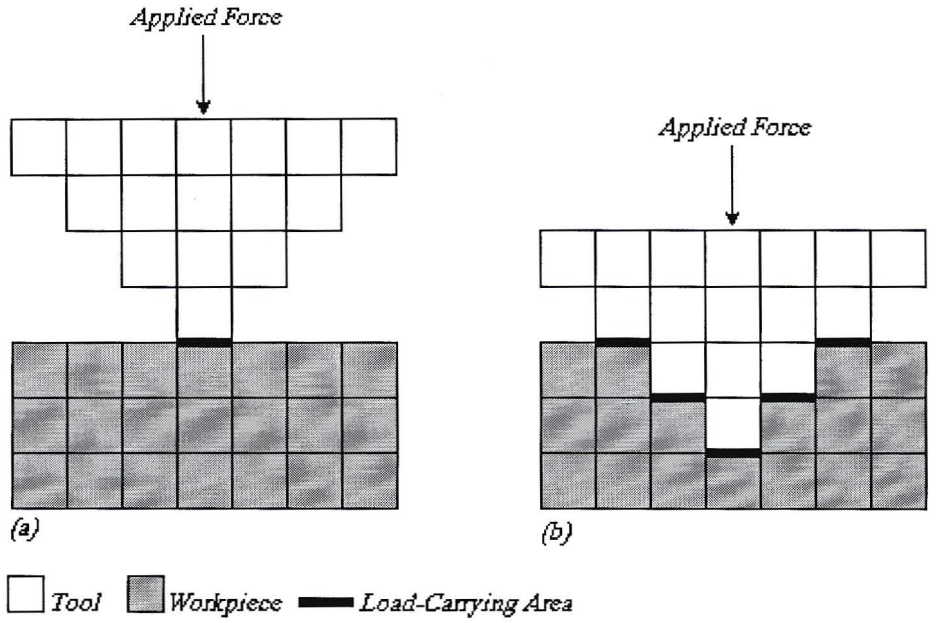


Figure 4.13 Cross-sectional area of the tool and workpiece, the load carrying area represented by the thick black line.

Algorithm 4.3 Calculating the Load Carrying Area

INPUT *Tool and Workpiece HMDS*

$A_c = 0$

FOR EACH *conical*

z_0 = Vertical position of *conical* relative to the workpiece
x_0 = x position of *conical* in the horizontal position
y_0 = y position of *conical* in the horizontal position

FOR r FROM *0* TO (*Number of conical HMDS rows*)
FOR c FROM *0* TO (*Number of conical HMDS columns*)
IF ((z_0 - *conical*[r, c]) ≤ *workpiece*[$r + x_0$, $c + y_0$])
$A_c = A_c + 1$

RETURN A_c

■ End of Algorithm

In Algorithm 4.3 when any portion of the conical penetrates or touches the workpiece the A_c is incremented. Algorithm 4.3 is applied to a conical and workpiece at different tool height positions, to find the load carrying area.

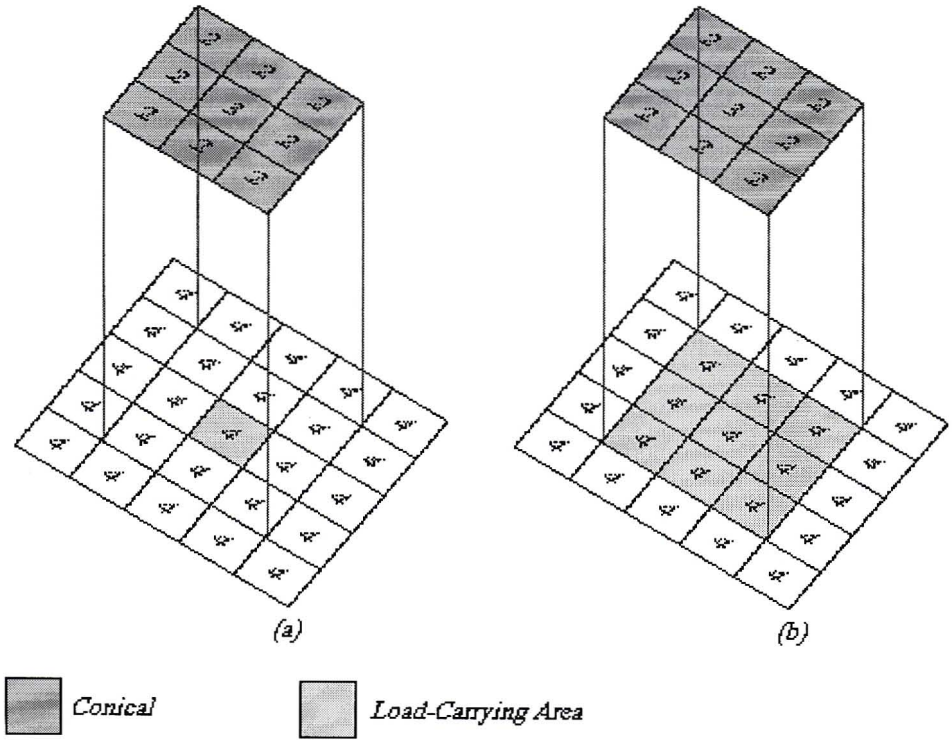


Figure 4.14 Calculated load-carrying area (a) Conical height position, z, is 7 units (b) z is 6 units

Figure 4.14 shows an A_c of 1 unit^2 at the conical height of 7 units. When the conical height changed to 6 units A_c is 9 units^2.

4.4.3 Penetration Depth

The penetration depth (p) is defined as distance the conical, on the tool, penetrates the workpiece (Section 3.2.4).

With reference to Figure 4.13(a) $p = 0$, and Figure 4.13(b) $p = 2$.

4.4.4 Worn Workpiece

The workpiece is worn when material is displaced from the groove. This displaced material is either ploughed to the groove sides or removed from the workpiece. The overlapping volume between the tool and workpiece is the worn volume of material from the workpiece. This event occurs at a single position in the path of the tool.

Algorithm 4.4 Worn Workpiece

INPUT *Tool* and *Workpiece HMDS*

FOR EACH *conical*

z_0 = Vertical position of *conical* relative to the workpiece
x_0 = x position of *conical* in the horizontal position
y_0 = y position of *conical* in the horizontal position

FOR r FROM *0* TO *(Number of conical HMDS rows)*
FOR c FROM *0* TO *(Number of conical HMDS columns)*

IF *(z_0 - conical[r, c]) < workpiece[r + x_0, c + y_0])*
workpiece[r + x_0 , c + y_0] = z_0 – conical[r, c]

RETURN worn *workpiece*

■ End of Algorithm

Algorithm 4.4 is applied to the two examples from Figure 4.14; the worn surface is displayed in Figure 4.15.

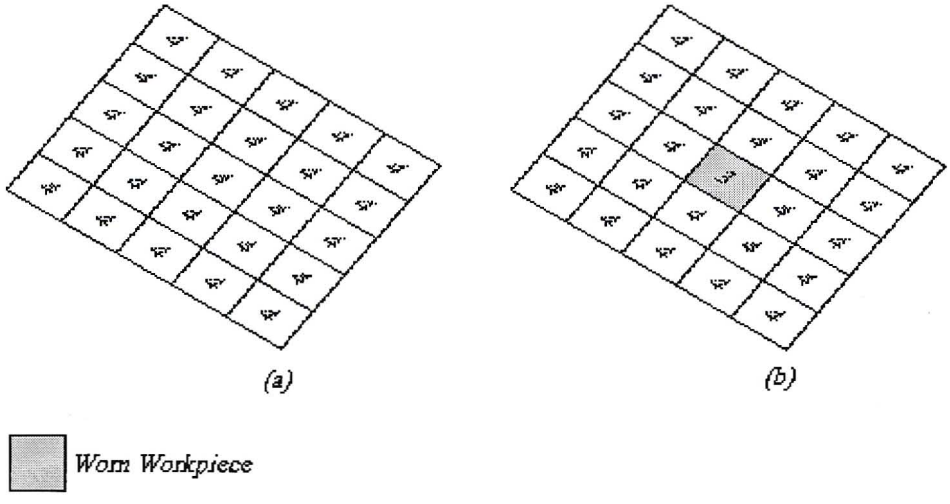


Figure 4.15 Worn workpiece (a) Conical y-position 7 units (b) Conical y-position 6 units.

The highlighted blocks in Figure 4.15 represent the portion of the workpiece that is worn and the numbers represent the new height values at that position. Figure 4.15(a) was not worn because the tool did not penetrate the workpiece.

4.4.4.1 Calculating Worn Volume

An advantage of using the HMDS is the simple calculation of the current volume.

$$V = \sum_{row=0}^{mR} \sum_{col=0}^{mC} workpiece[row][col] \qquad (4.6.)$$

where mR and mC are the numbers of rows and columns in the workpiece HMDS.

Consider the following two representations of the workpiece (Figure 4.16).

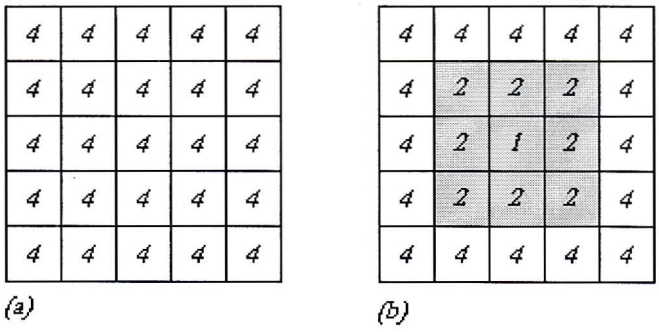


Figure 4.16 HMDS representation of the workpiece (a) Initial surface (b) Final surface after the surface was worn.

The worn volume (V_w) is the difference in volume between the initial volume (Figure 4.16(a)), V_i, and the finial volume (Figure 4.16(b)), V_f.

$$V_w = V_i - V_f \tag{4.7.}$$

With reference to Figure 4.16 and using Equation 4.7, the worn volume is 19 $units^3$.

4.4.5 Microploughing

Microploughing (Section 3.4.3) is the plastic deformation of the workpiece material. The volume of material ploughed to the sides of the grooves depends on the workpiece hardness.

In a pure microploughing state the total volume of material displaced from the grooves is deformed to the sides of the groove, without any material volume loss. The distribution of ploughed material around the grooves may be classified into two states. These states depend on the vertical and horizontal motion of the tool.

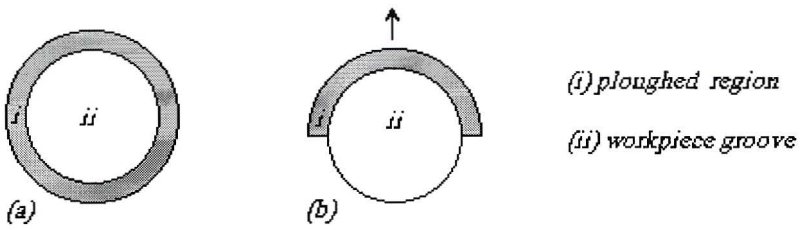


Figure 4.17 Top view of the ploughed material, material is displaced from the groove (ii) to the ploughed region (i). (a) Vertical state, VS (b) Horizontal state, HS

When the conical penetrates the workpiece vertically, material is displaced equally around the groove (Figure 4.17(a)), this is known as the vertical state, VS. The horizontal state, HS, material is displaced to the sides of the groove in the direction of the horizontal motion (Figure 4.17(b)). HS has ploughed material around half of the groove (Figure 4.18).

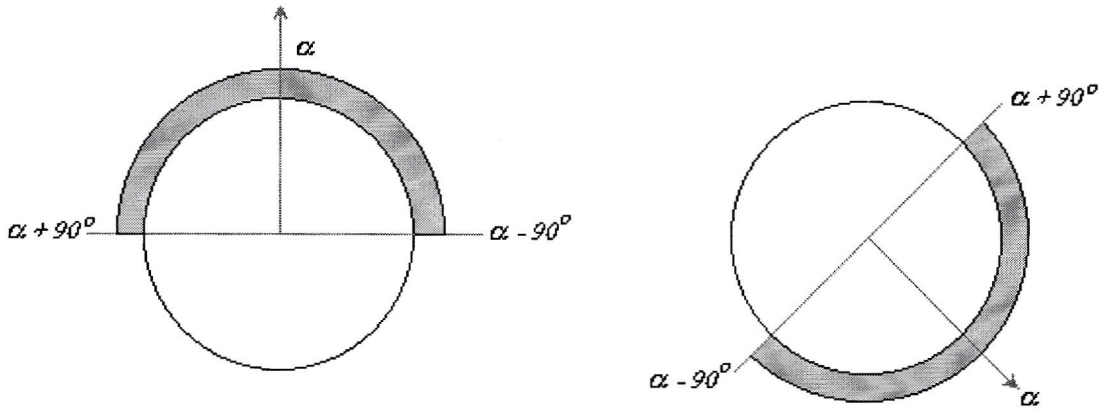


Figure 4.18 Geometric representation of the HS.

Algorithm 4.5 simulates the material ploughed to the side of the groove for a single conical on the tool.

Algorithm 4.5 Microploughing

INPUT conical and tool position

workpiece HMDS and its dimensions

volume, V_t, of material displaced from the groove
radius, r_g, of the groove formed by the conical
percentage, P_d, of displaced material that is ploughed

Calculate the radius, r, of the torus, using V_t, r_g and P_d. (see Section 2.3.4)

Outer torus radius, R = inner torus radius + $r*2$

IF there is ploughing
FOR x FROM $-R$ TO R
FOR y FROM $-R$ TO R
{ IF point (x, y) is found on the workpiece surface

IF point (x, y) is found in the ploughing region

DEPENDING on the ploughed state

ADD displaced material to ploughed region on the workpiece to form the torus shape.

}

RETURN workpiece

■ End of Algorithm

Consider the shape of A1 and A2 in Section 3.4.3. Algorithm 4.5 simulates microploughing by assuming the shape of the ploughed material to be the top half of a torus (Section 2.3.4). The torus provided a simple volume and surface equation that saved computer process time.

4.4.6 Tool Simulated Movements

The two different motions experienced by the tool are the horizontal movement, HM, and the vertical movement, VM, as in Figure 4.19. Together these two motions can position the tool anywhere in 3D space.

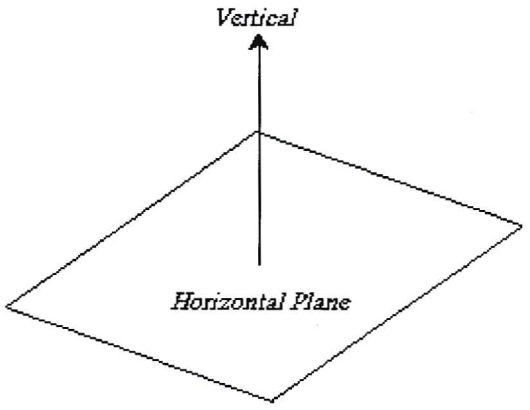


Figure 4.19 Horizontal and Vertical Motion

4.4.6.1 Horizontal Movement

HM is the combination of the tool spinning and moving back and forth. The simulation representation of the disc is a rectangle; the path travelled by the conical was also converted to cater for this change. This movement is responsible for traversing the workpiece. The disc rotation converts to a straight line, combined with the disc forward and backward motion (Section 3.5.3) resulted in an *Oscillating Wave* path (Figure 4.20).

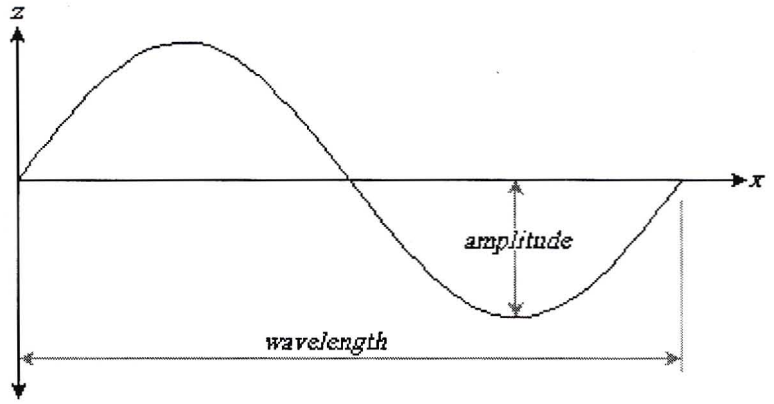


Figure 4.20 Path the Tool Travels in the Simulation

This path has two variables: amplitude and wavelength. The wavelength is related to the speed: of the rotating disc, and the forward and backward motion. Changing either speed results in a different wavelength. The distance the tool travels between any two consecutive direction changes, in the forward and backward motion, effects the amplitude.

$$z = A\cos\left[\frac{2\pi x}{w}\right] \tag{4.8.}$$

In equation 4.8, x and z refer to a position on the horizontal plane (Figure 4.20), A is the amplitude of the wave and w is the wavelength.

4.4.6.2 Vertical Movement

This VM raises and drops the tool, which redistributes the contact area between the two bodies.

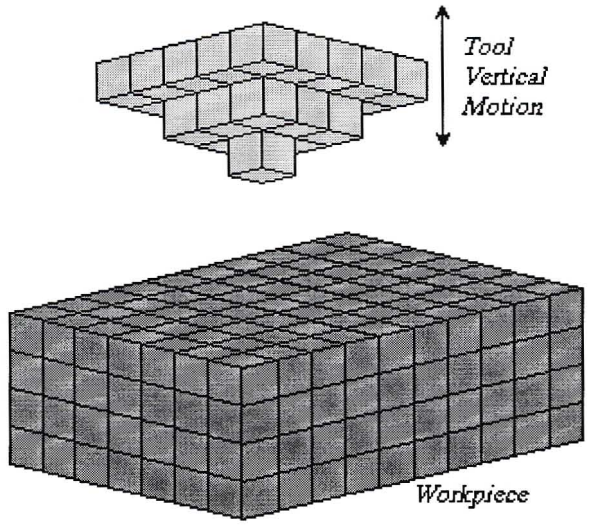


Figure 4.21 Tool's vertical motion

4.4.7 Simulation Flowchart

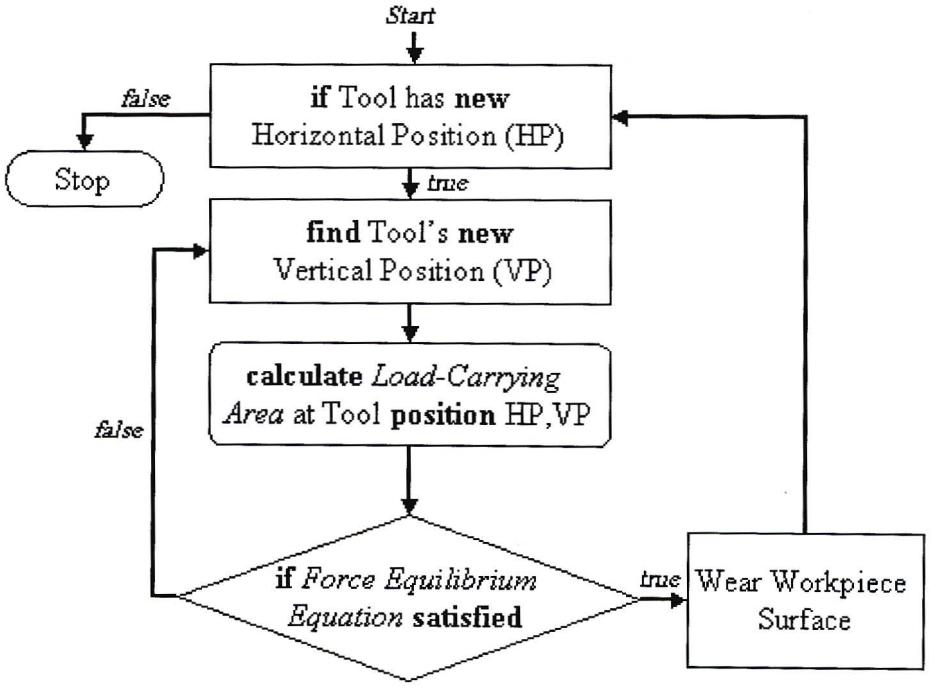


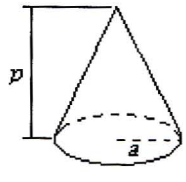
Figure 4.22 Flowchart Simulation

Figure 4.22 shows the top-level of the TBAW simulation. The first decision box outputs the new horizontal position of the tool. Thereafter the tool's vertical position is chosen, using the modified search method (Section 2.3.7.3). Given any position of the tool (HP and VP), the A_c[5] between the tool and workpiece is calculated. If the A_c satisfies the *FEE* the workpiece is worn[6] else a new position is calculated. The new shape of the worn workpiece surface depends on: the exact position of the tool relative to the workpiece, the shape of the initial workpiece surface, and the size, shape and position of each conical on the tool. This process is repeated until the entire horizontal path is traversed.

4.5 Balancing Problem

The balancing problem deals with satisfying the FEE (Equation 3.1). This problem is totally dependent on the resolution of the objects in the simulation. High object resolution results in greater accuracy and the FEE is easily balanced (E.g. FEE is dependent on load carrying area, with a higher object resolution the load carrying area would be more accurate). In what follows the abrasive is represented as a cone so the balancing problem can be demonstrated.

4.5.1 Geometric Formulas



$$\text{Cone Surface Area} = \pi a\sqrt{a^2 + p^2}$$
$$= \pi p^2\sqrt{2} \text{ if } (a = p)$$
$$A_c = \text{Projection of surface area normal to the Applied Force}$$
$$= \pi p^2$$

Figure 4.23 Cone

[5] A_c was calculated using Algorithm 4.3
[6] Worn surface used Algorithm 4.4

A tool with one abrasive particle penetrates (Section 3.2.4) the workpiece at a depth of p, therefore $A_c = \pi p^2$. Increasing the penetration depth to $(p+1)$ then A_c increases to $\pi(p+1)^2$. In other words A_c is proportional to the square of the penetration depth ($A_c \alpha p^2$).

A_c is used in the FEE. A change in the penetration depth results in a quadratic change in A_c.

Suppose the penetration depth p is discretized as $p = m\,\delta p$, so that $A_c = \pi m^2 \delta p^2$. Therefore, the FEE is not exactly satisfied, since the applied force does not necessarily equal the restoring force HA_c for some value of m. As δp becomes smaller, the FEE can be satisfied to a better approximation. The penetration depth, p, is chosen so that A_cH is as close as possible to F.

In the TBAW process the tool consists of multiple abrasive particles (cones). Therefore $A_c \alpha np^2$, where n is the number of cones. Changing the penetration depth, p results in a larger change in the A_c. Thus the balancing problem is more severe when dealing with multiple abrasive particles.

4.6 Summary

The new TBAW simulation has been discussed. All algorithms, flowcharts and problems have been outlined.

Then next chapter focuses on the implementation of this simulation and compares the results with other simulations.

Chapter 5 Simulation Methodology and Results

5.1 Introduction

The previous chapter looked at how a computer simulation can be applied to the TBAW process. This chapter presents an implementation of a TBAW simulation and the results obtained from it. The experiments that were run and the procedures used are also described.

The parameters of the TBAW process that influence the wear process are investigated. Comparisons with results of previous authors are made.

5.1.1 Simulation Algorithm Implementation

The simulation used for the experiments is implemented in Borland C++ Builder, *Deitele & Deitel [14]*, version 5.0 (Build 12.34). A Microsoft Windows 98 platform is used on a Pentium Celeron 333 with 160M Ram, and 8Mb AGP graphics card. All coding is object oriented.

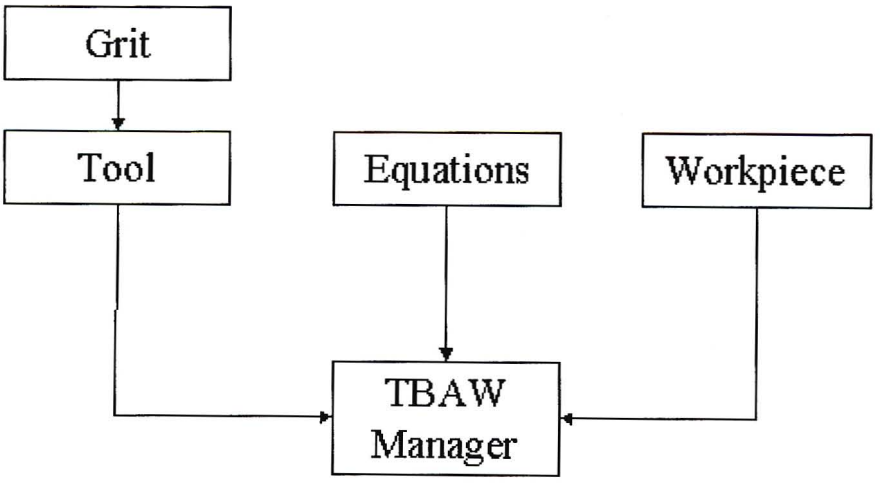


Figure 5.1 Simulation's Object Orientated Design

The implementation combines algorithms from chapters two, three and four, and creates the simulation. There are three components to the simulation: digital object construction, object interaction and three-dimension rendering. Digital object construction and object interaction are described in Chapter 4. Please refer to Appendix D to learn how to use the TBAW Simulation.

The simulation program includes user input, graphs, program indicators and a 3D rendering. User inputs are the parameters values (see Section 5.1.2) used to run the simulation. The program generates different graphs to help understand the wearing process. The indicators in the program help the user interpret the current events of the simulation. The 3D rendering is an implementation of a 3D graphics engine to display the wearing process taking place on the HMDS bodies. This 3D rendering uses the OpenGL modelling library, as described in Chapter 2.

The simulation is designed to form a working environment that could be used to test different workpiece material, tools and interactions between the two bodies by providing different parameters. It is also easy to add new representations of abrasive shape.

5.1.2 Parameter Domain

Various parameters for the different bodies and events can be experimented with. These bodies include the tool, conical, workpiece and their interaction. The list of parameters and their description is shown in Table 5.1.

Bodies/Events	**Description**
Conical	Abrasive Representation
Tool	Collection of Conical
Workpiece	Wearing Body
Interaction	Wear process

Table 5.1 Simulation parameters

Tables 5.2-5.5 show the numerical value ranges that are used to demonstrate the ability of the simulation to produce realistic abrasive wear results. All of the parameter values should be used within the specified range (due to the processing power available).

5.1.2.1 Conical

The conical parameters used in the simulation are listed in Table 5.2. These parameters were described more fully in Section 4.2.1.

Parameters	**Values**
Diameter Size	10-200 μm
Standard Deviation	2-100 μm
Tip Radius	0-100 μm
Cone Angle	45-120^o

Table 5.2 Conical parameters

5.1.2.2 Tool

The tool is constructed from conicals in a base, this process is documented in Section 4.2. The parameter values domains are listed in Table 5.3.

Parameters	**Values**
Base Width	10-500 μm
Base Length	10-500 μm
Total Volume of conical	101-2 000 000 μm^3

Table 5.3 Tool parameters

5.1.2.3 Workpiece

The parameter values for the workpiece (see Section 4.3) are shown in Table 5.4. The *initial surface roughness* has two parameters; number of holes and maximum hole depth, depending on these two variables a rough initial workpiece is created which is measured

in R_a (*R_a is defined as the arithmetic mean deviation of the surface height from the mean plane through the surface profile.*).

Parameters	**Values**
Width	10-500 μm
Length	10-500 μm
Hardness	0-maximum workpiece hardness N/mm^2
Initial Surface Roughness	0-20 R_a
Number of holes	0-300
Maximum hole depth	0-50 μm

Table 5.4 Workpiece parameters

5.1.2.4 Interaction

These interaction parameters in Table 5.5 are used in the wear process (see Section 4.4).

Parameters	**Values**
Load	0.05-10 N
Maximum Workpiece Hardness	0-10000 N/mm^2
Tool Path Amplitude	0-200 μm
Tool Path Wavelength	0-200 μm
Tool Distance Travelled	0-100000 μm

Table 5.5 Interaction parameters

5.1.3 Simulation Experimental Methodology

There are many different parameters that can be adjusted when executing a particular simulation run. If an attempt is made to try every variation, the number of trials required becomes prohibitive. However, it is still interesting to look at different combinations of parameters, as there can be relationships between them. The parameter values considered are largely from *Jacobson et al* and *Jiang et al* [1, 8].

The runs are categorised as follows:

- The first set of generated data demonstrates a *single simulation run* on some defined set of parameters. The purpose of this single simulation run is to understand the methodology of obtaining simulation results.
- The aim of the next simulated runs is to validate the simulation with well-known grit size effect results from the *literature* [1, 8], therefore varying the conical diameter, and applied normal force or conical tip radius.
- Once the simulated results resembled the literature, they were then compared with *experimental* data.

The program assumes all unit of measurement in micrometer (μm), unless otherwise stated. This is for easy comparison between the simulation, literature and experimental results.

5.1.4 Discrepancies with the literature

This simulation has been compared with previous literature and experimental results. There are discrepancies due to the different scales used.

In Jacobson el. at. [1] the Dimensionless Load, D_L, is defined as

$$D_L = \frac{F_N}{HA_c} \tag{5.1}$$

where F_N is the applied normal force, H is the workpiece hardness and A_c is the load carrying area. D_L is dimensionless, since

$$\dim(D_L) = \frac{N}{\frac{N}{mm^2} mm^2} \tag{5.2}$$

This simulation uses the same units as the literature, but the scale of the dimensionless load could not be used. Consider a D_L value of $0.18\text{x}10^{-3}$ that is used in the literature, this value is converted to the corresponding F_N by using Equation 5.1. From [1] H is 700

N/mm^2 and A_c is $500\mu m$ x $500\mu m$, (which is limited by the maximum resolution our machine, PC, can cope with) therefore F_N is 0.0315N. This F_N is used in the FEE, Section 3.2.1. In order for the simulation to produce the same valued results as [1], it would need to use similar values for H, A_c and F_N. The first problem is that this F_N is below the minimum range for the simulation input, and the second is the Balancing Problem. The third problem is that the average size of A_c used in this simulation is $200\mu m$ x $200\mu m$. A larger load carrying area increases the time needed to run each simulation, on the PC specified (see Section 5.1.1), and is therefore limited by processing power.

This simulation uses a larger Force on a smaller Load Carrying Area, therefore a larger Dimensionless Load. Thus this simulation shows larger values than in the literature. But the **trends** can be compared.

5.2 Single Simulation Run

The aim of this simulation run is to demonstrate the method of obtaining simulation results, after the tool has moved a given distance. A rough initial workpiece surface is chosen, and pure microcutting is assumed without any ploughing.

5.2.1 Parameter Settings

All parameter settings for this run are shown in Table 5.6.

Workpiece	
Width	*260 μm*
Length	*200 μm*
Hardness	*700 N/mm^2*
Initial Surface Roughness	*3.64 Ra*
Tool	
Base Width	*200 μm*
Base Length	*200 μm*
Total Conical Volume	*234 929μm^3*
Interaction	
Normal Force	*0.8N*
Amplitude	*30 μm*
Wavelength	*30 μm*
Tool Distance Travelled	*1 200 μm*
Conical	
Mean Conical Diameter	*50 μm*
Conical Density	*325 mm^{-2}*
Conical Tip Radius	*10 μm*
Cone Angle	*90^o*

Table 5.6 Parameter settings for the single simulation run

5.2.2 Results

The results from the tool sliding over workpiece, in this single simulation run, are shown in Table 5.7. These test values were recorded after the tool has travelled its total sliding distance. For this entire study all measurement shown is the average of three simulation runs.

Test Methodology	**Values**
Dimensionless Wear Rate	0.0080 $(\mu m)^3/(\mu m * \mu m^2)$
Wear Coefficient	0.0100 $\mu m^3/(N * \mu m)$
Surface Roughness	1.47 R_a

Table 5.7 Single simulation run results

Another important measurement from this run is the rate at which the surface roughness changes as the sliding distance increases; this is illustrated in Figure 5.2.

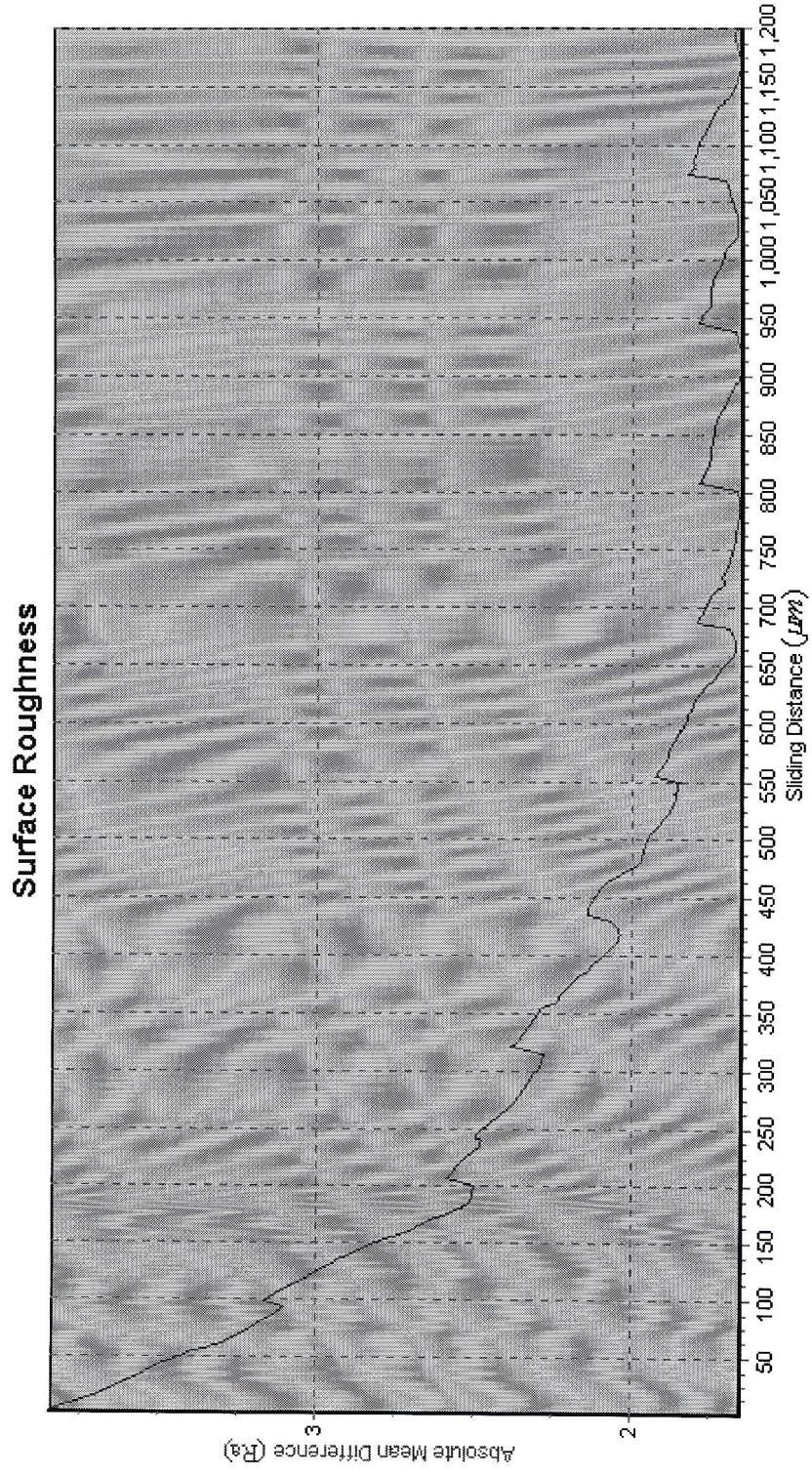


Figure 5.2 Surface roughness vs. Sliding distance from the simulation.

Figure 5.2 clearly shows that the surface roughness decreases as the sliding distant increases. R_a decreases sharply thereafter maintaining a steady value. This effect can be related to the TBAW application of polishing. The final R_a value, after the tool has completed its
total sliding distance, was largely dependent on the tool's mean conical size.

5.3 Literature Comparison

The wear is largely influenced by the size of conical; most of the research in this field investigated the conical size effect [1, 8, 18]. The tool slides over the workpiece a forward distance of 200 μm, thereafter the wear rate, wear coefficient and surface roughness is measured. These simulation results are compared with the literature data.

5.3.1 Parameter Settings

All of the simulation runs in this Section 5.3 use parameter values listed in Table 5.8, unless otherwise stated. Microploughing is assumed.

Workpiece	
Width	*260 μm*
Length	*200 μm*
Hardness	*700 N/mm²*
Initial Surface Roughness[7]	*0 Ra*
Tool	
Base Width	*200 μm*
Base Length	*200 μm*
Interaction	
Amplitude	*30 μm*
Wavelength	*30 μm*
Tool Distance Travelled	*200 μm*
Conical	
Conical Tip Radius	*10 μm*
Cone Angle	*90°*

Table 5.8 Conical size effect parameters

5.3.2 Conical Density

This Section 5.3.2 is based on verifying the density of conical on the tool. Density (Γ) is the number of conicals per unit area. The effect the conical size has on the density of conicals on the tool is investigated.

In order to make this comparison, the simulation generated tools with different mean conical size. At each mean conical size three tools were generated, the average density

[7] The initial workpiece surface is perfectly smooth

value was then calculated. The density of conicals *vs.* conical size is demonstrated in Figure 5.3 and Figure 5.4.

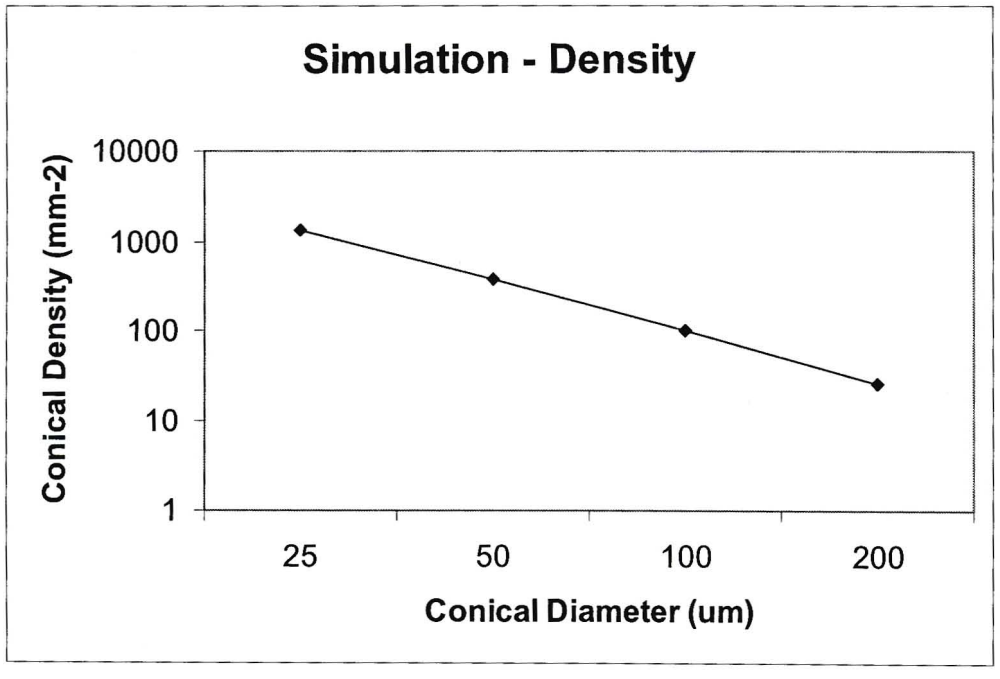


Figure 5.3 Conical density vs. conical diameter size

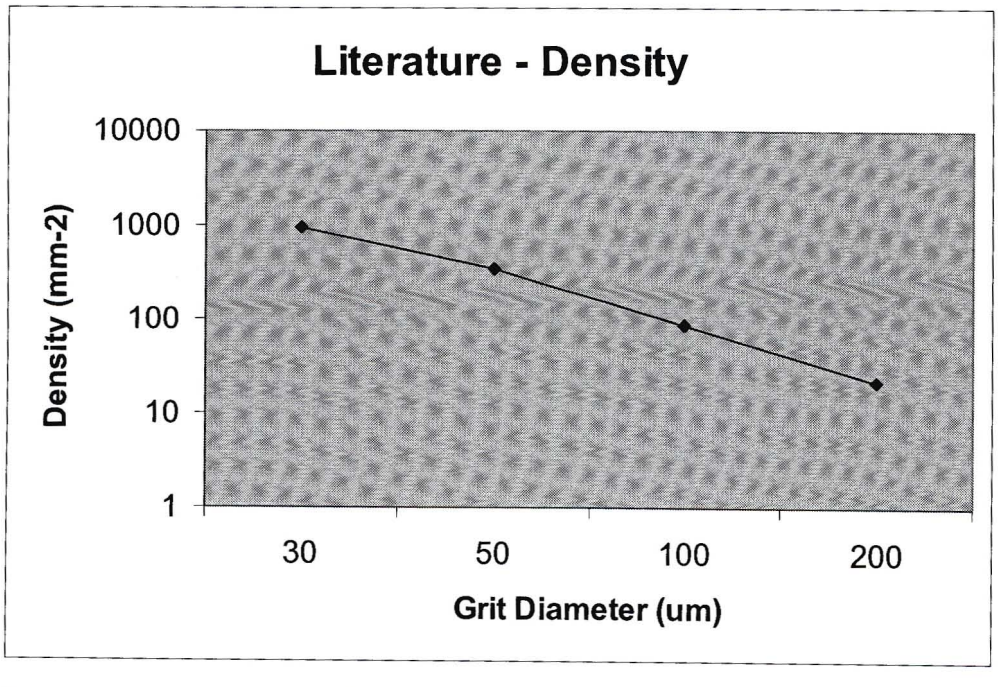


Figure 5.4 Density vs. grit diameter size from Jacobson et al. [1]

The variation of particle density with conical size is given in Figure 5.3, which shows the density is inversely proportional to the square of the mean conical diameter, for similar packing conditions. This correlates with the literature results, shown by *Jacobson et al* [1] in Figure 5.4. All simulation runs that follow assumes the tool to have the maximum density of conicals for a given conical size distribution.

5.3.3 Wear Rate

The wear rate (Q) is measured as in Section 3.6.2. Simulation runs were generated using a conical size of between 0 and 200 μm, at approximately 25 μm intervals. At each interval four different dimensionless loads (see Section 5.1.4) were used. The Q results for the simulation runs are shown in Figure 5.5. This is compared with results from *Jacobson et. al.* [1], as in Figure 5.6. Unfortunately the actual values from the Wear Rate graph, Figure 5.6, in [1] cannot be compared with the Simulation Wear Rate graph, Figure 5.5. The reason for this inaccuracy is due to a number of factors; smaller workpiece and tool area, smaller dimensionless load and the Balancing Problem, see Section 4.5. Therefore only graph **trends** are discussed.

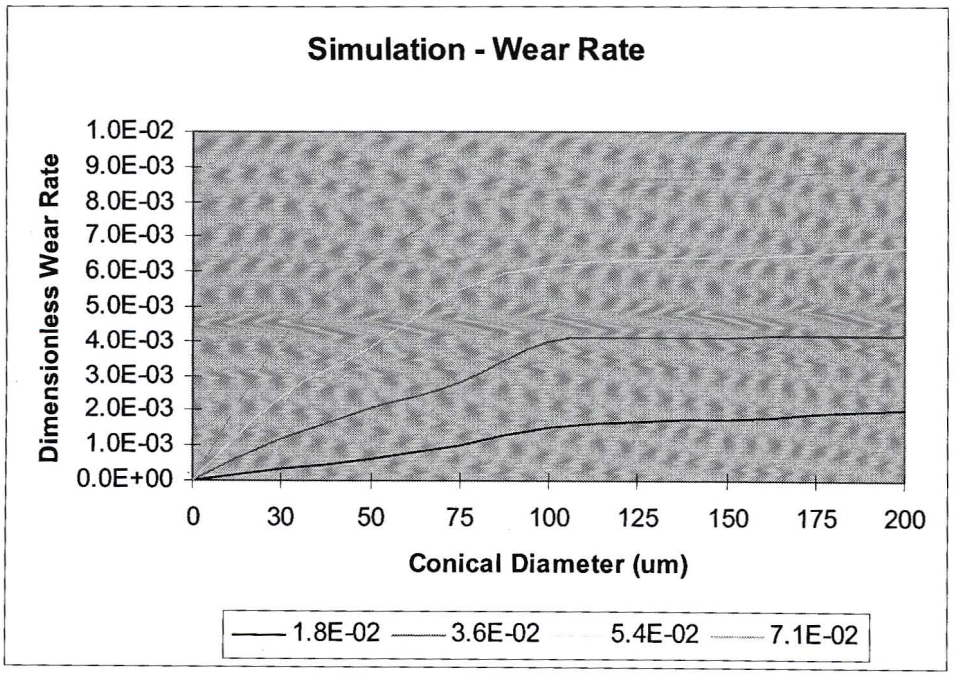


Figure 5.5 Wear Rate vs. Conical diameter size for different dimensionless loads.

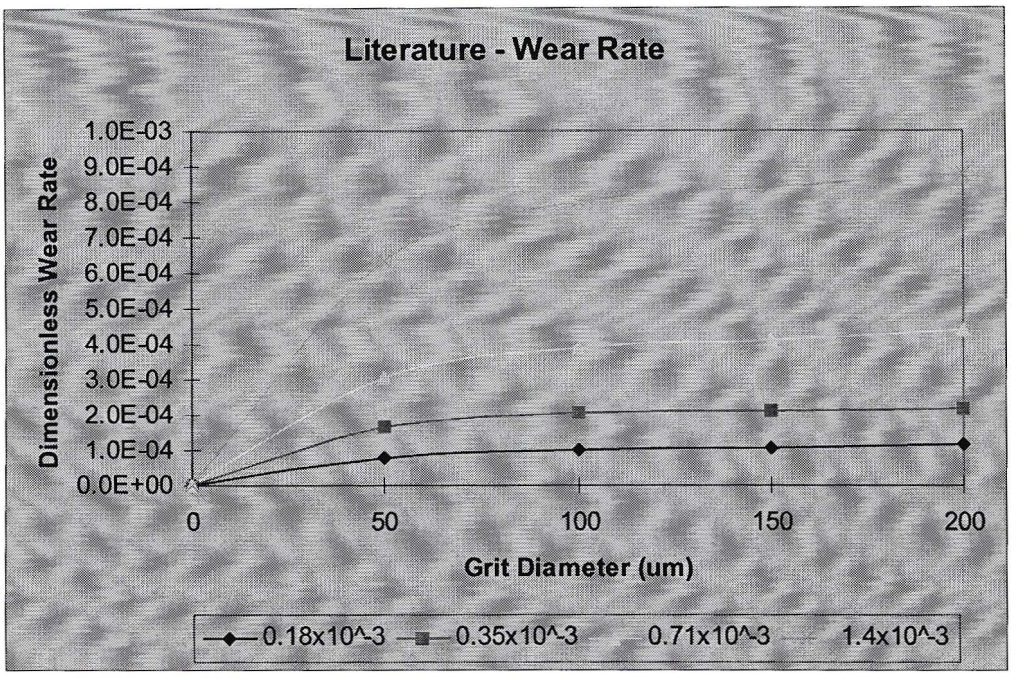


Figure 5.6 Wear rate vs. Grit diameter size from Jacobson et al [1] at different dimensionless loads.

The simulation curves, in Figure 5.5, reproduce the true general behaviour as to size dependence. As the conical diameter increases Q increases rapidly until a critical conical size is reached. Q then becomes independent of conical diameter or increases slightly. F is proportional to Q (with constant conical size). The slope depends on the normal load. The critical conical diameter for transition to a constant Q with increase in conical size is about 100 μm. The simulation, Figure 5.5, reproduces the grit-size effect in abrasion, which was demonstrated by *Jacobson et al* [1], Figure 5.6. The dimensionless wear rate produced by the simulation, Figure 5.5, is one order of magnitude higher than the *Jacobson et al* wear rate [1], Figure 5.6, due to the factors mentioned in Section 5.1.4.

5.3.4 Wear Coefficient

Section 3.6.3 discusses the variables and units of the wear coefficient. The wear coefficient is measured at different conical size, approximately 25 μm intervals between 0 and 200 μm, and using four different dimensionless loads. The wear coefficient results generated by the simulation, Figure 5.7, are compared with *Jacobson et al* [1] Figure 5.8.

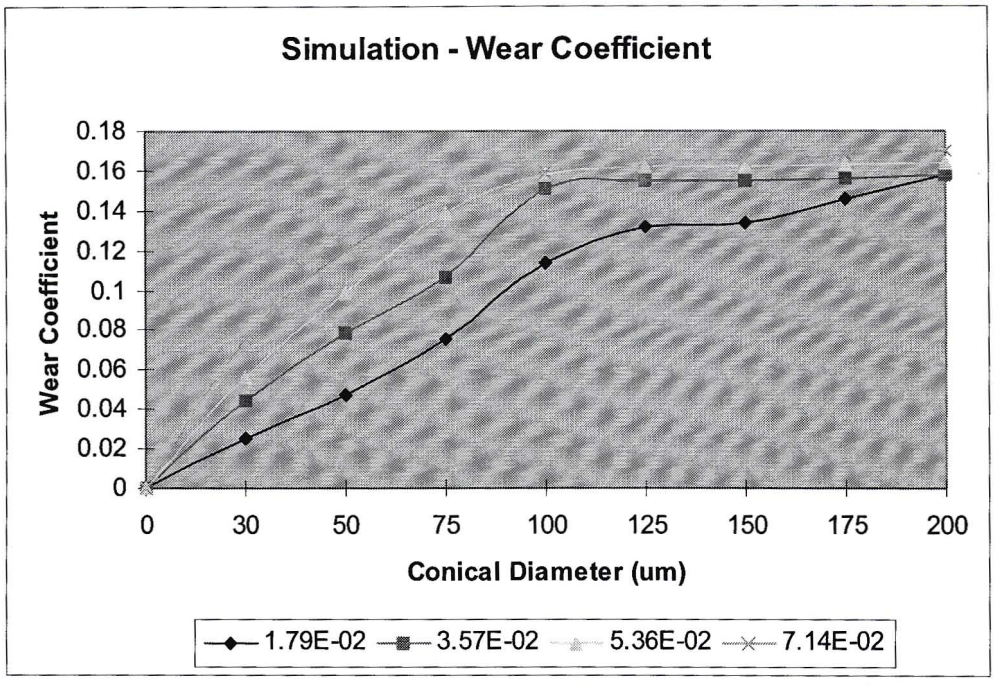


Figure 5.7 Wear coefficient vs. Conical diameter size for different dimensionless loads.

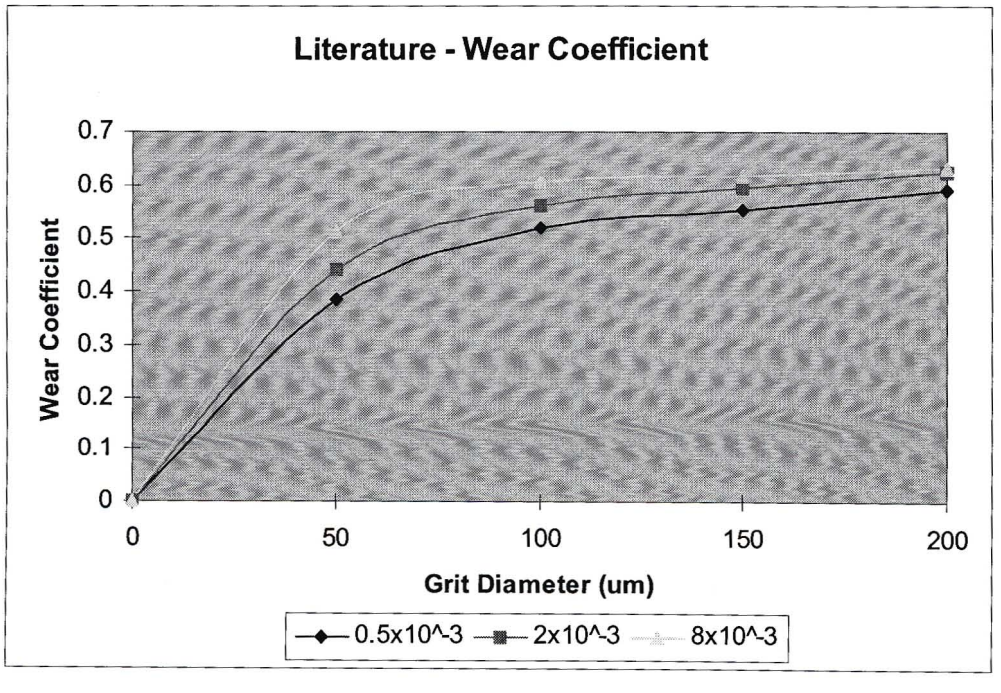


Figure 5.8 Wear coefficient graph from Jacobson et al. at different dimensionless loads [1].

In Figure 5.7, as the conical diameter increases the wear coefficient rapidly increases until a critical conical size is reached, and thereafter the wear coefficient remains constant or increases slightly. The slope depends on the normal dimensionless load. The simulation, Figure 5.7, reproduces the well-known grit size effect in abrasion, demonstrated by *Jacobson et al [1]* in Figure 5.8.

5.3.5 Surface Roughness

The influence of grit size and load on the resulting surface roughness is often of prime interest in the TBAW grinding and polishing applications. The tool is not a smooth object and therefore sliding over the workpiece surface, with an applied normal load, the workpiece is deformed and worn. The worn workpiece is investigated and measured in R_a. The initial workpiece surface (see Section 3.6.1) is assumed to be smooth, with R_a = 0. The results generated by the simulation are compared with results from *Jacobson et al* [1]. The simulation recorded R_a at varying conical size with different dimensionless load. Figure 5.9 and Figure 5.10 show the surface roughness *vs.* conical size.

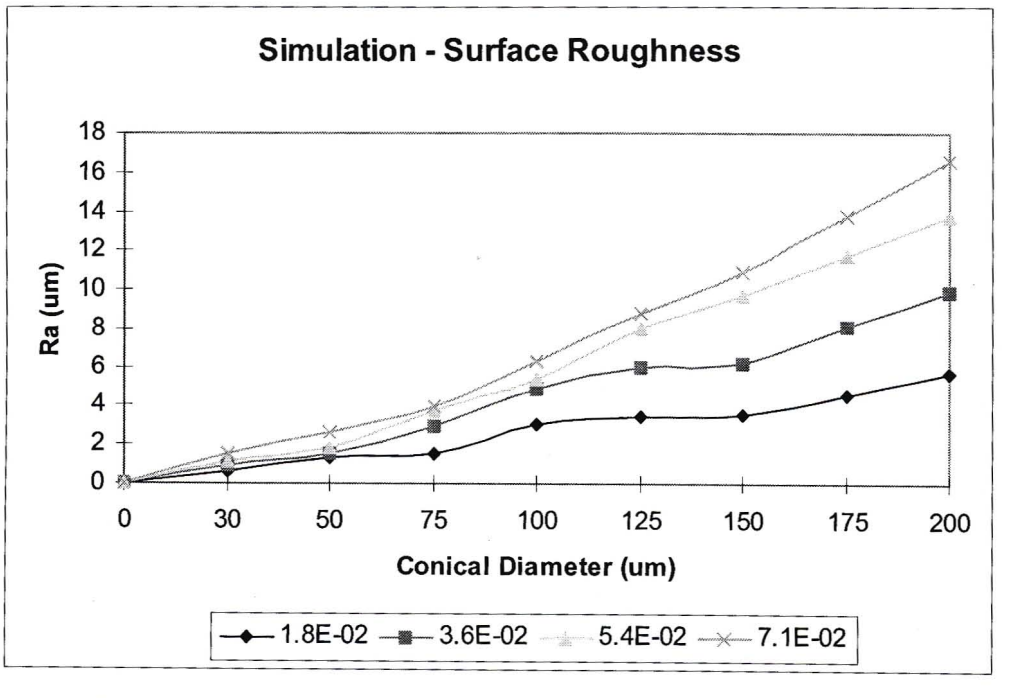


Figure 5.9 Surface roughness vs. Conical diameter size for different dimensionless loads.

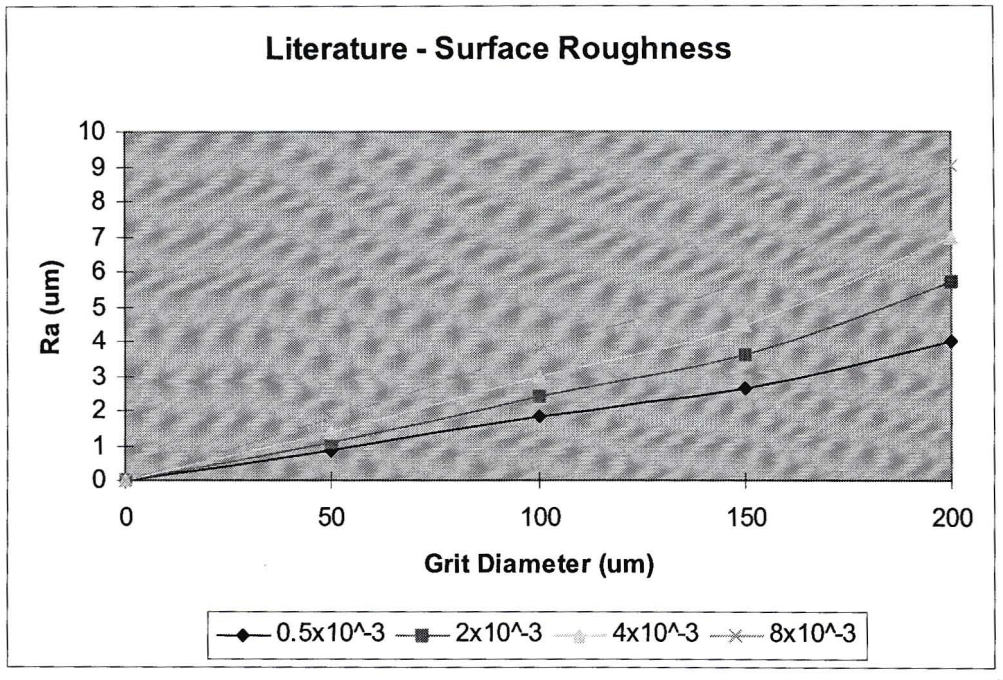


Figure 5.10 Surface roughness graph from Jacobson et al [1] for different dimensionless loads.

The general trends of the simulation result, Figure 5.9, show resemblance to the *Jacobson et al* [1] results, Figure 5.10. Larger conical size produces rougher workpiece surfaces, assuming constant force. The greater normal load, assuming constant conical size, also results in rougher workpiece surfaces.

5.3.6 Wear Rate and Tip Radius

The method of measuring the wear rate results is the same as in Section 5.3.3, except for the difference of the conical tip radius being a parameter and not the applied normal force, F. This section is based on investigating the relationship between conical tip radius, grit size and wear rate. Results generated from the simulation were compared with the *Jiang et. al.* [8] results. The importance of using tip radius as a parameter, is to understand the effects of the different abrasive particle shapes; sharp and rounded abrasives. The simulation, Figure 5.11, assumes constant $F = 1\ N$.

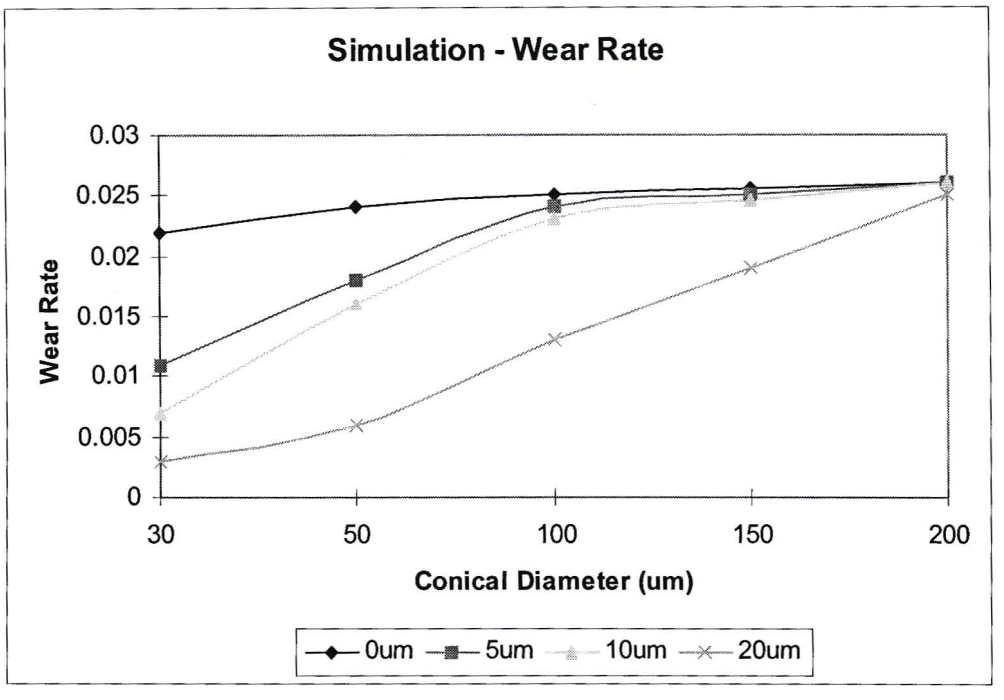


Figure 5.11 Wear rate vs. Conical size at varying tip radius

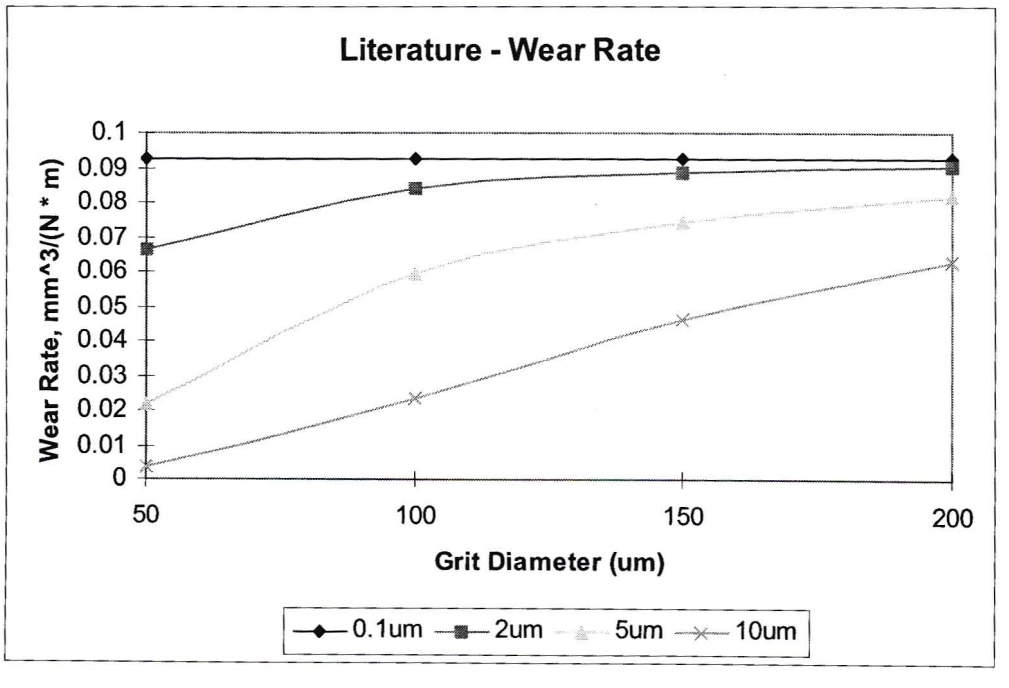


Figure 5.12 Wear rate vs. Conical size at varying tip radius, from Jiang et. al. [8].

The critical grit size phenomenon[8] is apparently observed, see Figure 5.11, for abrasive particles with tip radius greater than and equal to 0 μm. For particles with larger tip radius at the small grit size, zero wear rate is predicted by the simulation. With very sharp abrasive particles (r = 0 μm), wear rate is independent of grit size. High wear rate is produced with sharp particles compared with blunter ones. With increase in tip radius, wear decreases for a given size of conical particles and the critical grit size increases. This compares well with *Jiang et. al.* [8], Figure 5.12.

5.3.7 Discussion

The simulation was compared with those of other researchers with excellent results being shown. The conical density, wear rate, wear coefficient and surface roughness values have been shown to produce realistic results with the grit size effect. This forms the foundation of validating the results from the simulation.

Due to the simulation choosing a random conical position and conical size distribution on the tool, this simulation does not produce the smooth graphs found in the literature [1, 8, 9, 18]. The results produced by the simulation are more attuned with reality. Appendices A, B and C show the error due to the randomness of the simulation.

5.4 Experimental Comparison

Experimental data was obtained from *Sin et al [18]*, to verify the simulation results.

Commercially pure nickel was used as the workpiece, with a diameter of 6.35 mm. Abrasion experiments were conducted on the pin-on-disk set-up. The workpiece held in a loading arm followed a spiral track on a 20 cm diameter abrasive tool and therefore always passed over fresh abrasives. The total sliding distance was 4 m. The normal load was applied by a dead weight. Grit size ranging between 0 and 200 um was used and the normal load was varied between 5 and 40 N. [18]

[8] The grit size has little or no influence on the wear rate for small conical tips radius, at a constant normal force

5.4.1 Parameter Settings

The simulation data used in Section 5.3 is also compared with the experimental data, and uses the parameters of Section 5.3.1.

5.4.2 Wear Rate

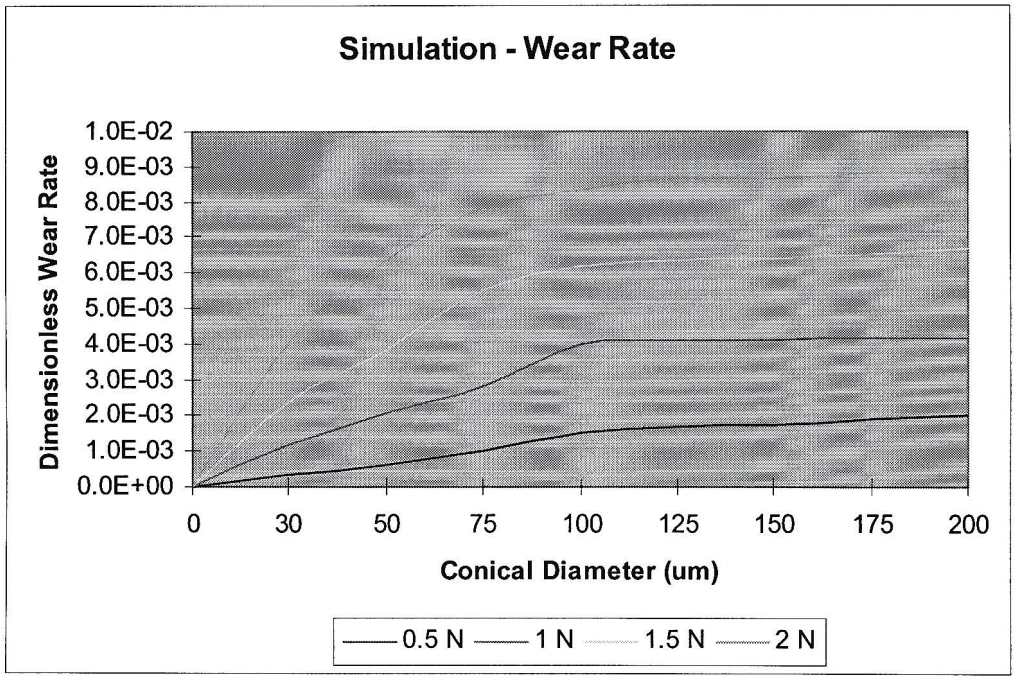


Figure 5.13 Wear Rate vs. Conical diameter size for different applied normal forces.

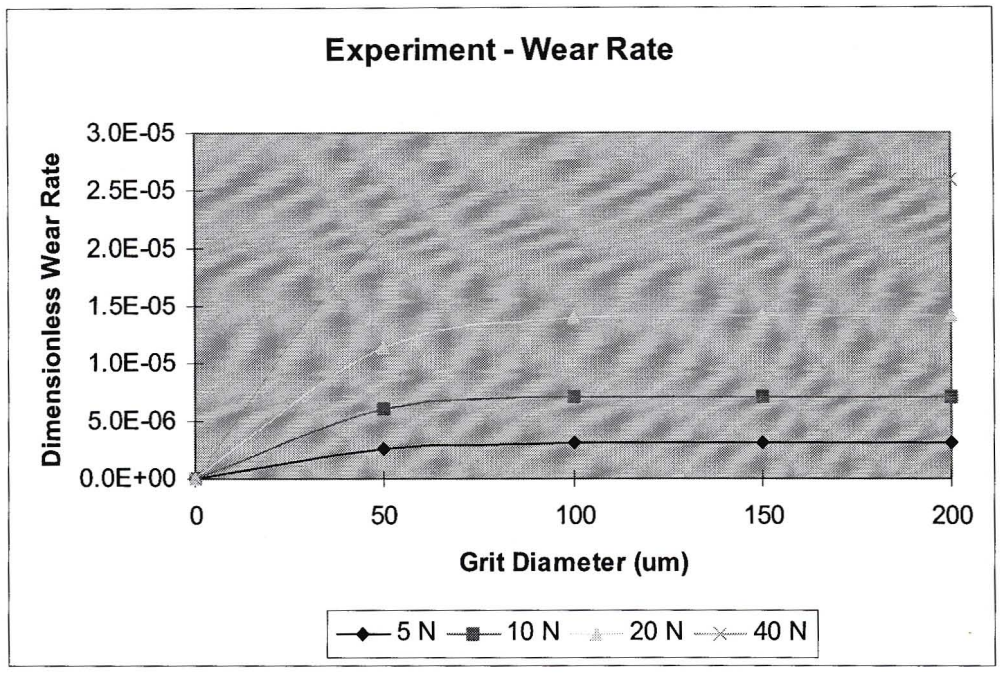


Figure 5.14 Wear Rate vs. Conical diameter size for different applied normal forces from Sin el at. [18]

The simulation data, Figure 5.13, trends do agree with the experimental data, Figure 5.14. Both the graphs show the grit size effect. The major contributing factor for the simulation wear rate being greater than the experiment wear rate for the same workpiece, is the value of the applied force per area, Table 5.9.

	Simulation Pressure (SP) N/mm^2	**Experiment Pressure (EP) N/mm^2**	**SP / EP**
	12.50	0.16	77.50
	25.00	0.32	77.50
	37.50	0.65	58.13
	50.00	1.29	38.75
Average (SL / EL)			**62.97**

Table 5.9 Pressure Difference

Notice from Table 5.9, the difference between the simulation pressure and experimental pressure is about a order of magnitude of two. Due to the problem described in Section 5.1.4, the smallest simulation load chosen is 12.5 N/mm^2.

5.4.3 Wear Coefficient

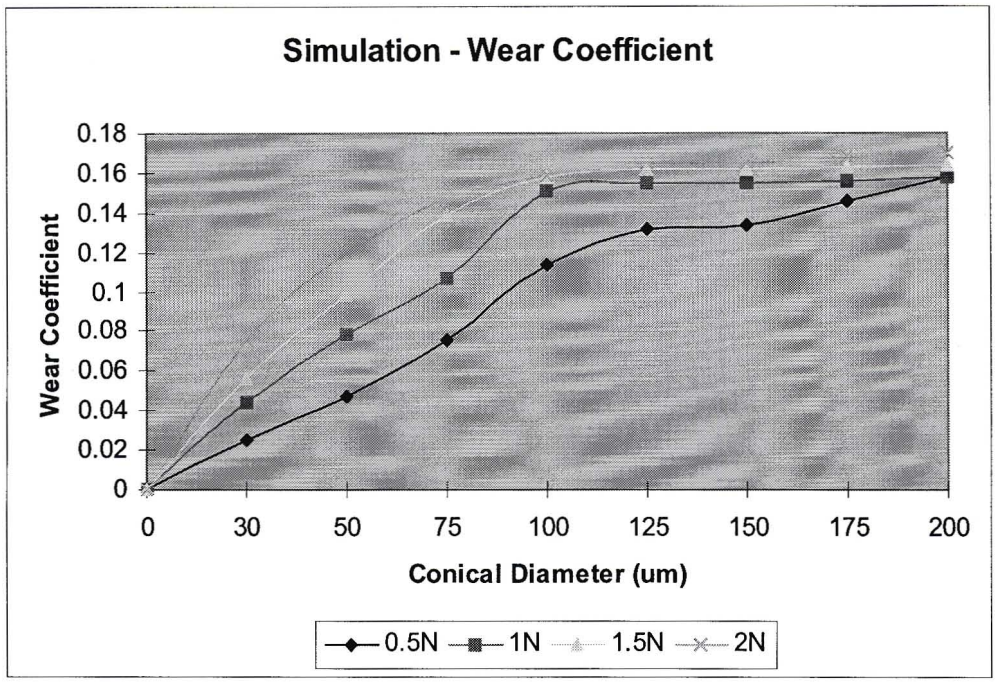


Figure 5.15 Wear coefficient vs. Conical diameter size for different applied normal forces.

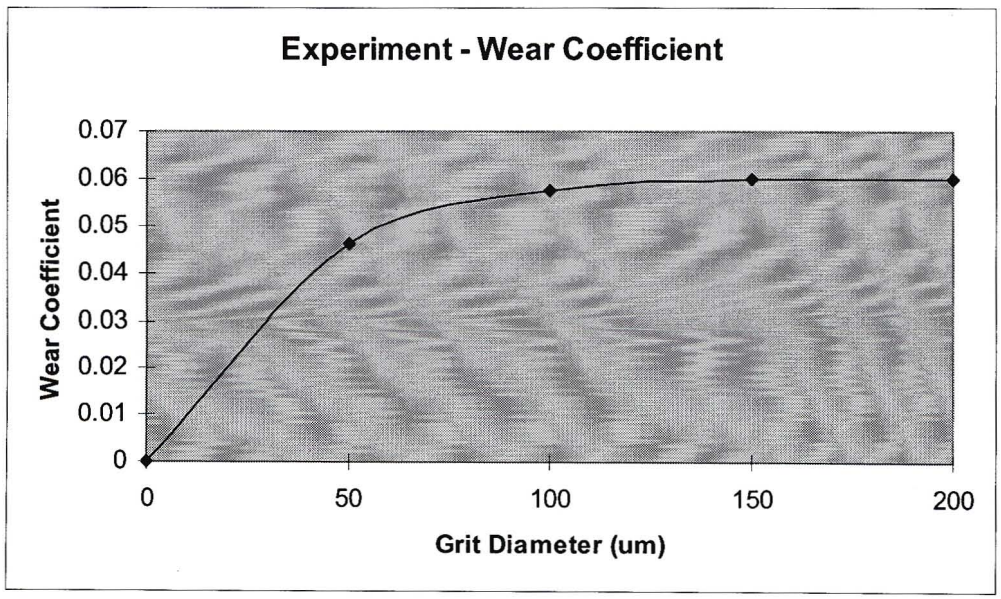


Figure 5.16 Wear coefficient vs. Conical diameter size from Sin et al. [18]

As in Section 5.3 the simulation, Figure 5.15, produces the grit size effect, which is also seen in the experimental data by *Sin et al* [18], Figure 5.16.

5.5 Summary

The simulation was implemented and simulation runs parameters were chosen. The simulation was then compared against those of other researchers with favourable results being shown but there is clearly room for improvement. In particular, selecting lower dimensionless load would seem beneficial so closer comparisons can be made with previous results. Increased processing power would surely enhance this simulation.

Chapter 6 Conclusion

The aim of this research was to investigate the application of a computer simulation to the two-body abrasive wear process. This aim was fulfilled by a review of the current literature on computer simulations of the two-body abrasive wear process, the implementation of a new computer simulation for the two-body abrasive wear, and a comparison of these new results with results obtained by previous authors using different modelling techniques and experiments.

A flexible computer simulation framework was developed in C++. This framework was designed to allow the effect of a number of different parameters to be investigated. The effect of a combination of two-body parameters was investigated, including grit size, tip radius, grit density and applied force on the tool at constant workpiece hardness (this implies that the amount of material microploughed is constant) and tool path. The wide range of possibilities resulted in over 1000 simulation runs being performed. The results (wear rate, wear coefficient and surface roughness) were encouraging as the implementation confirmed results obtained in previous research.

There is still plenty of research that can be performed concerning the application of the two-body abrasive wear simulation. This is clear from the wide variety of approaches to the problem that has been successful. It should be possible to use the techniques outlined in this thesis to tackle the lubricated two-body abrasive wear problem. The three-body lubricated and un-lubricated process should also be amenable to simulation techniques.

The two-body abrasive wear process is an important problem that leads to an understanding of the contact and interaction between two objects. The results from this research have provided valuable insight into the parameter that affects the abrasive wear process.

Appendix A. Wear Rate Error

The following graphs illustrate the similarities and difference in the wear rate values generated by the simulation. Four different mean conical sizes with three different forces were used. Each run was done five times, and the standard deviation was recorded. All simulation runs used the parameter settings from Section 5.3.1.

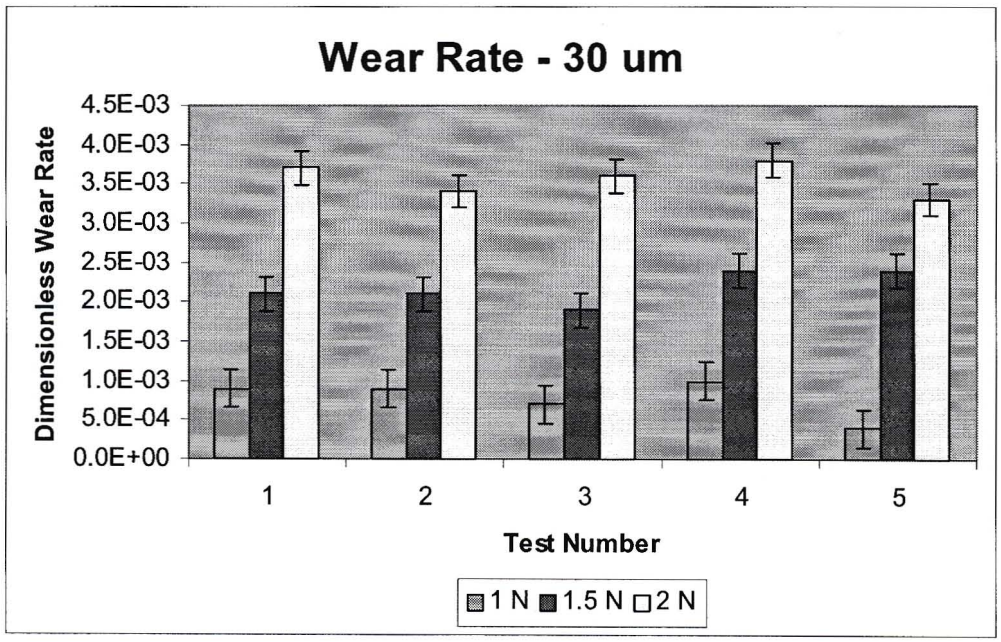


Figure A. 1. Wear rate at 30μm with varying force

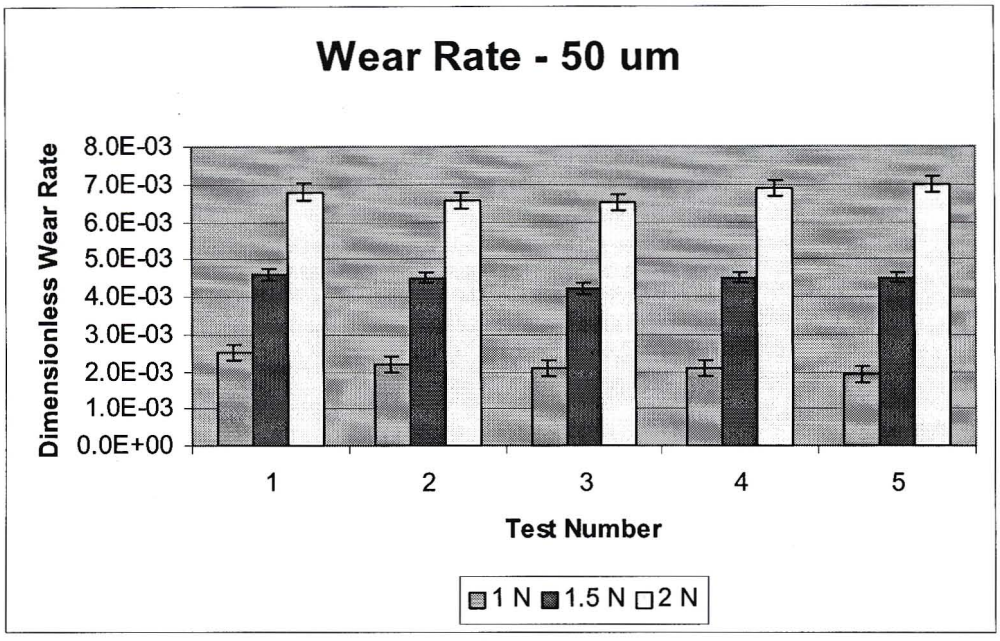


Figure A. 2. Wear rate at 50μm with varying force

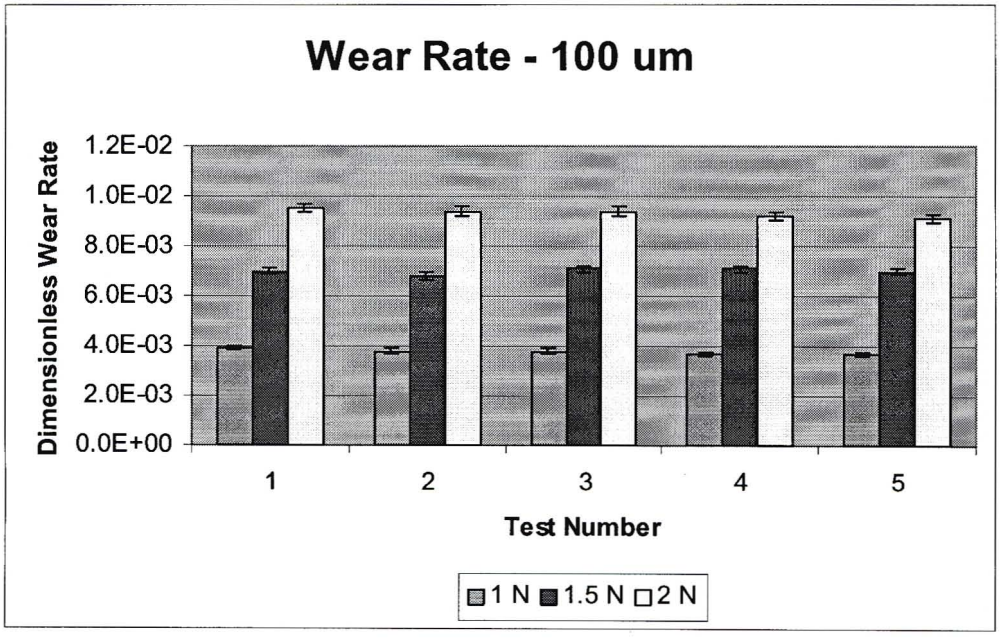


Figure A. 3. Wear rate at 100μm with varying force

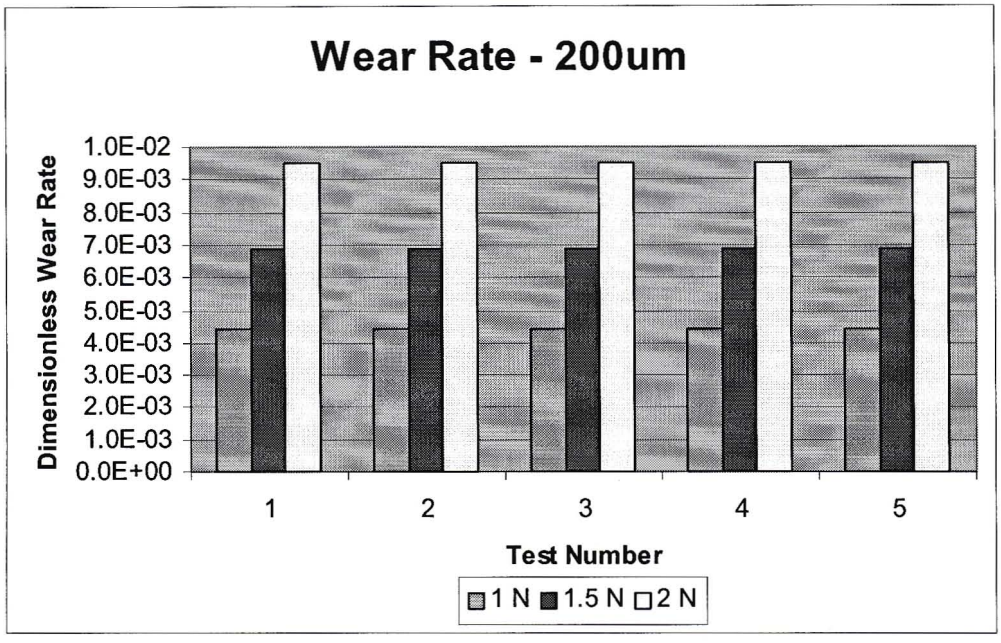


Figure A. 4. Wear rate at 200μm with varying force

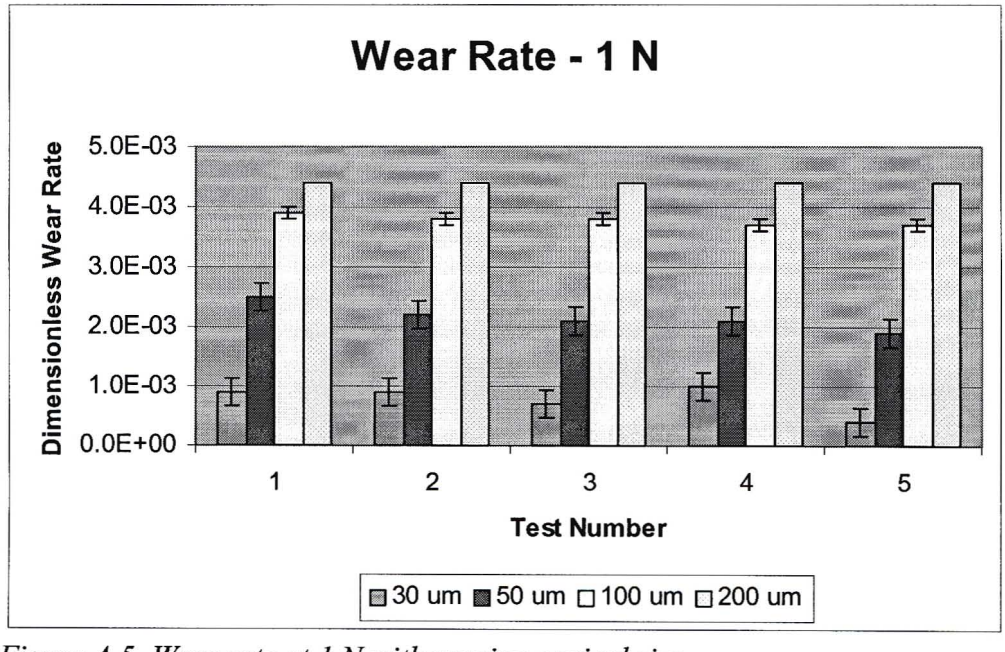


Figure A.5. Wear rate at 1 N with varying conical size

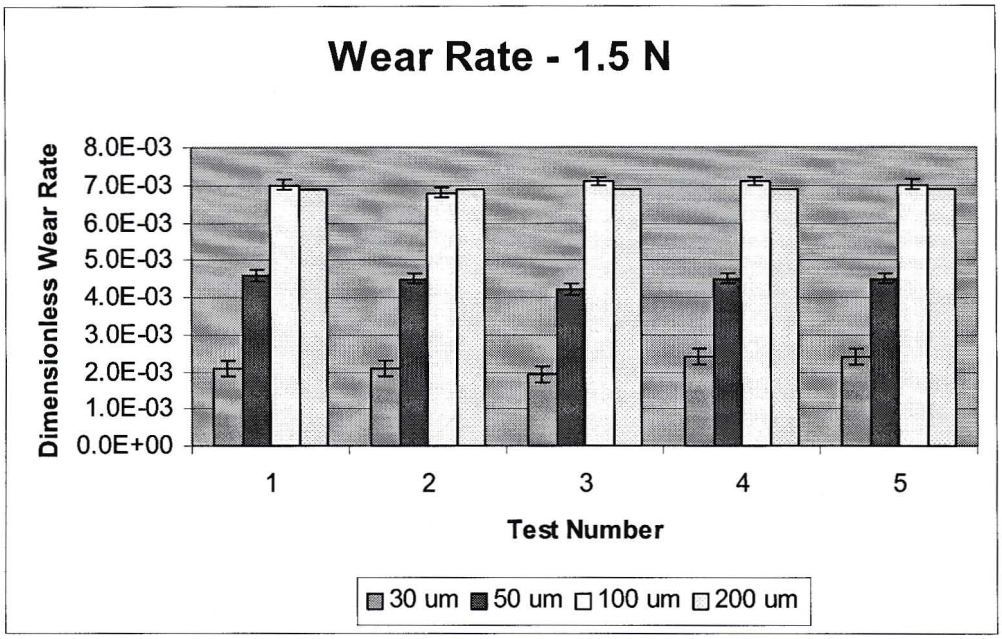


Figure A. 6. Wear rate at 1.5 N with varying conical size

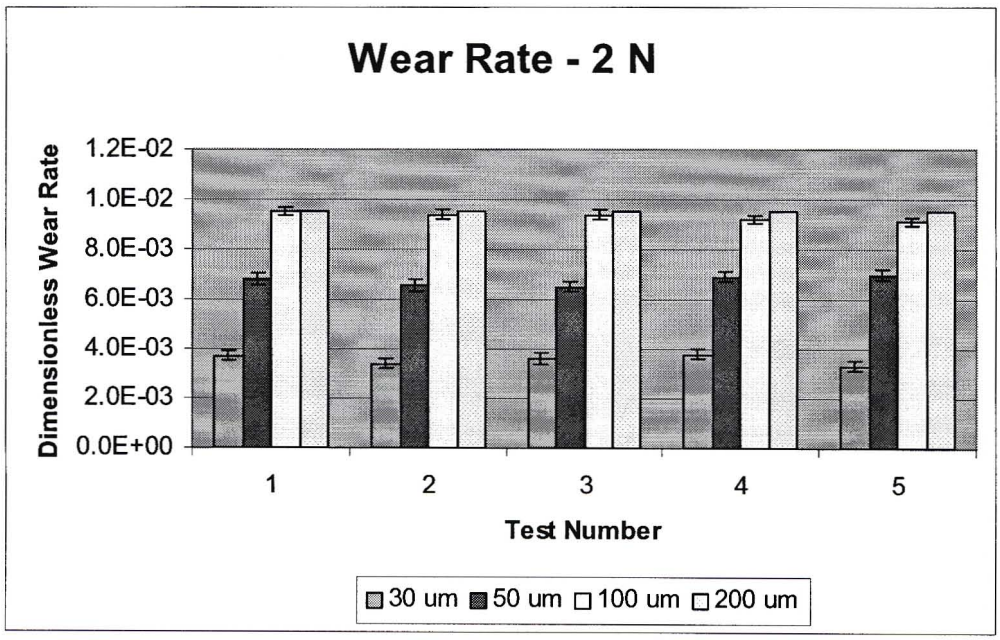
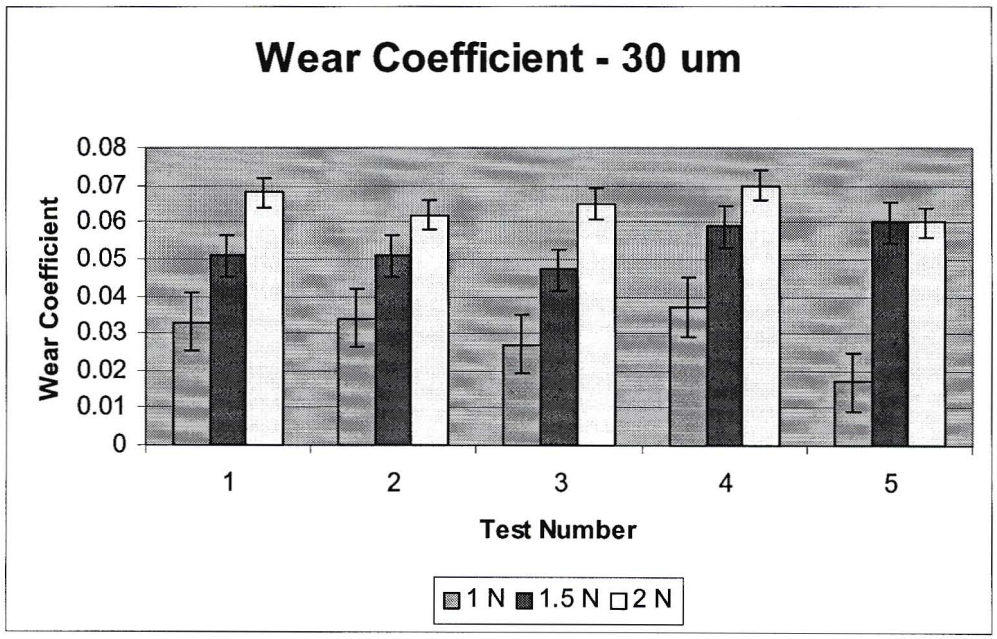


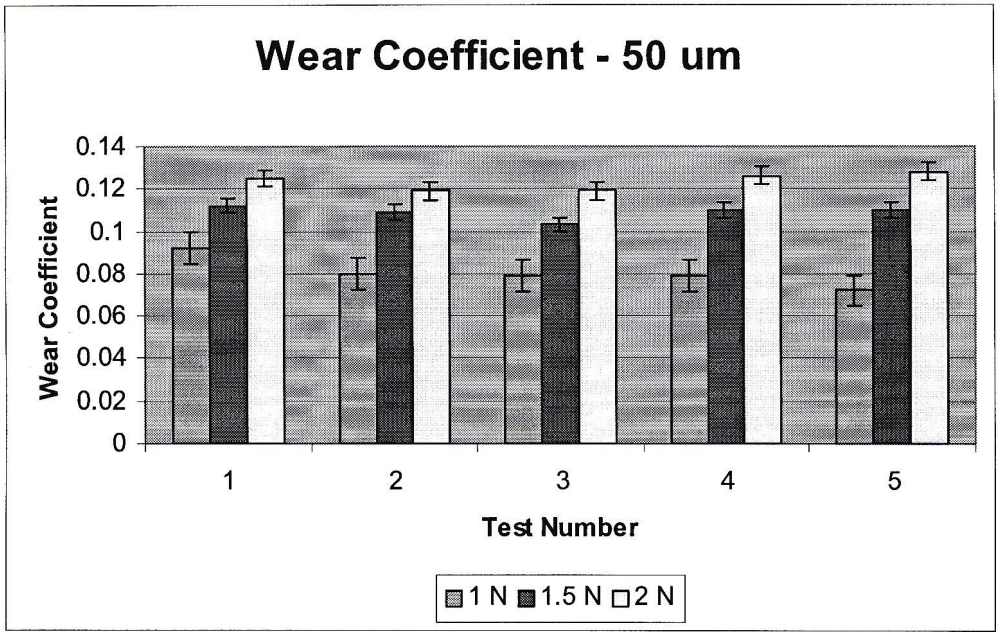
Figure A. 7. Wear rate at 2 N with varying conical size

Appendix B. Wear Coefficient Error

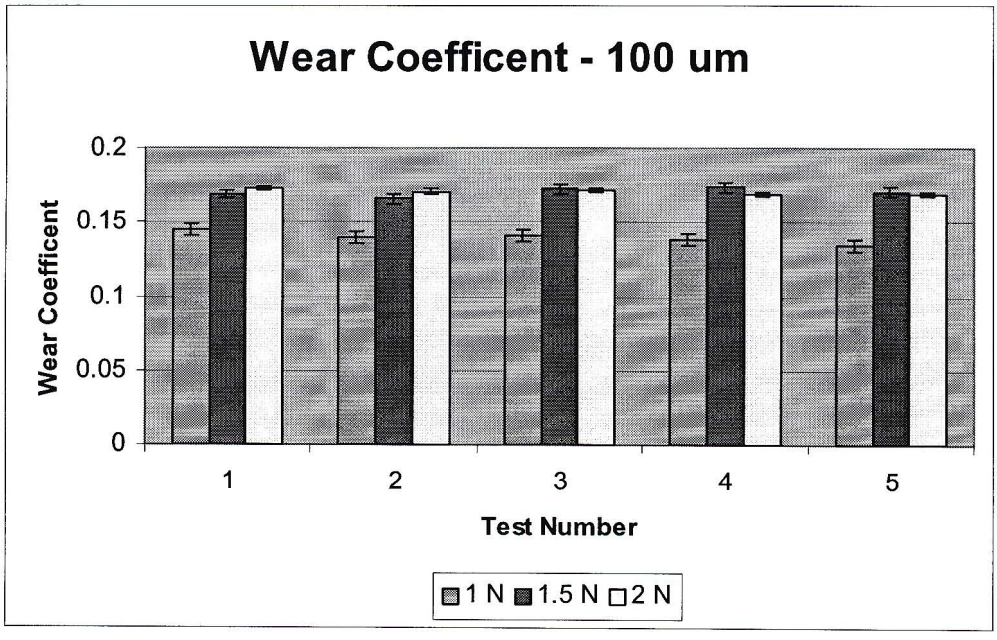
The following graphs illustrate the similarities and difference in the wear coefficient values generated by the simulation. Four different mean conical sizes with three different forces were used. Each run was done five times, and the standard deviation was recorded. All simulation runs used the parameter setting from Section 5.3.1.



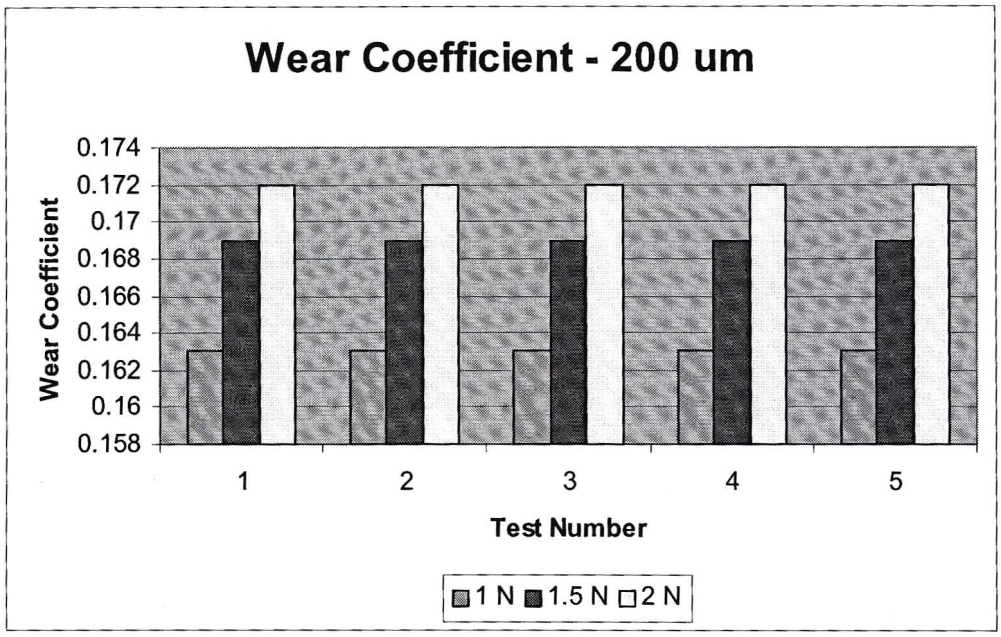
Appendix B. 1. Wear coefficient at 30μm with varying force



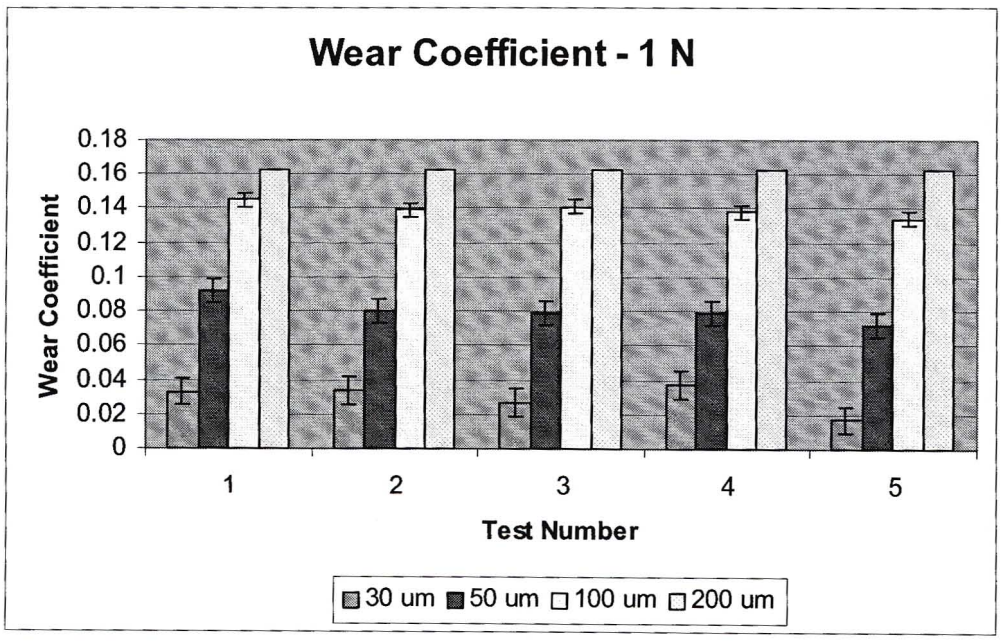
Appendix B. 2. Wear coefficient at 50μm with varying force



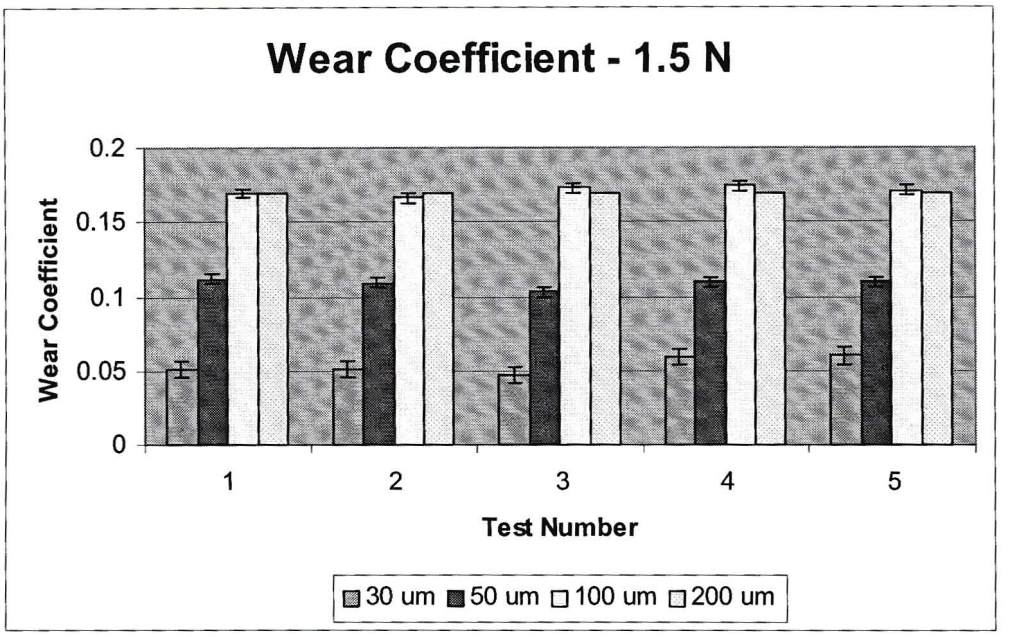
Appendix B. 3. Wear coefficient at 100μm with varying force



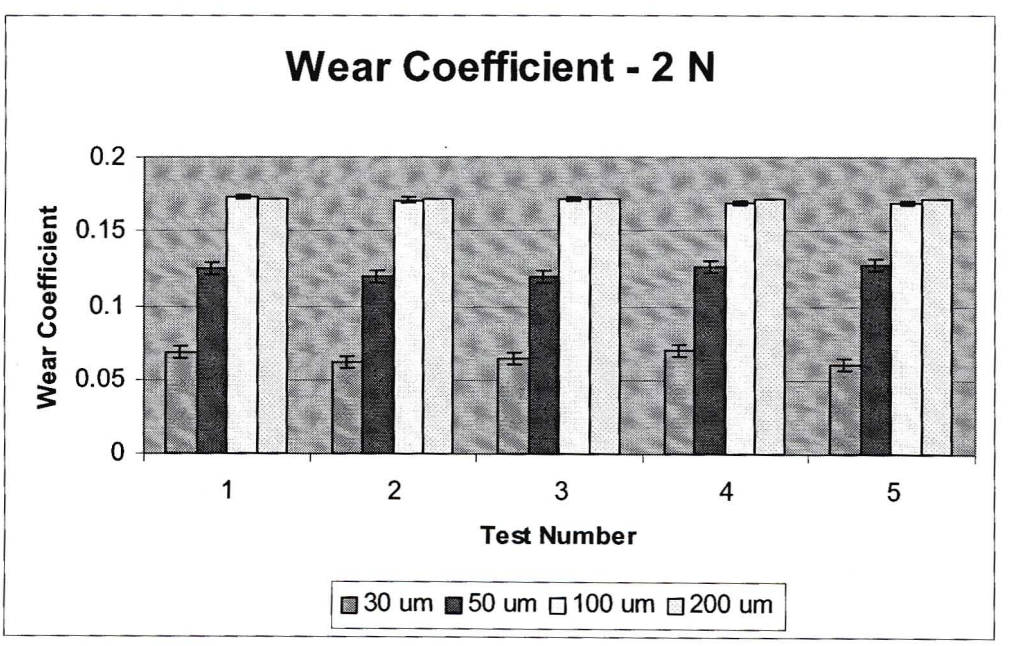
Appendix B. 4. Wear coefficient at 100μm with varying force



Appendix B. 5. Wear coefficient at 1 N with varying conical size



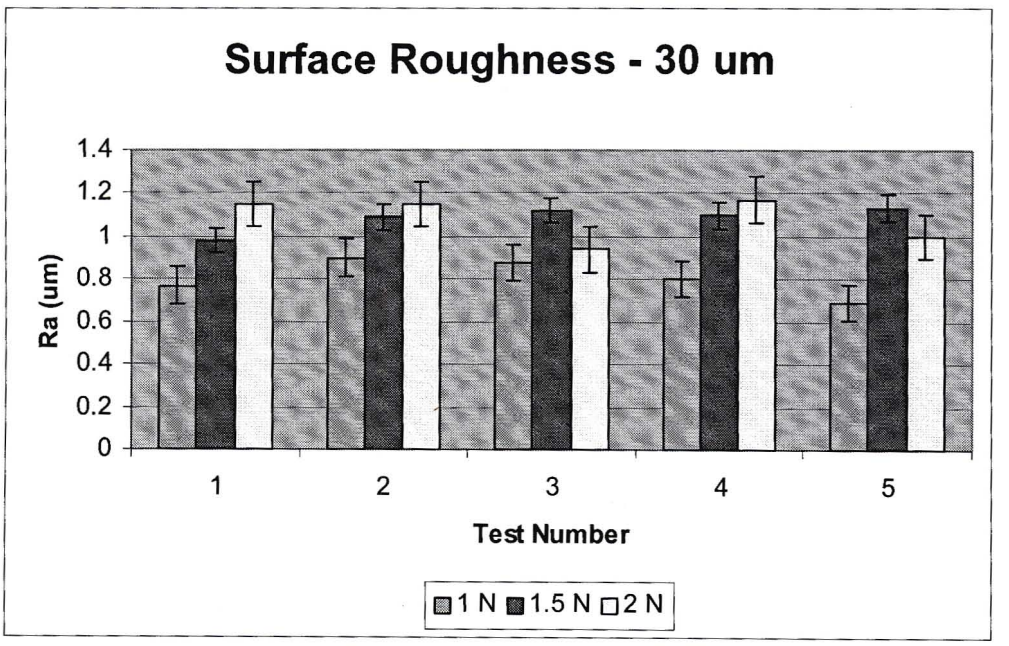
Appendix B. 6. Wear coefficient at 1.5 N with varying conical size



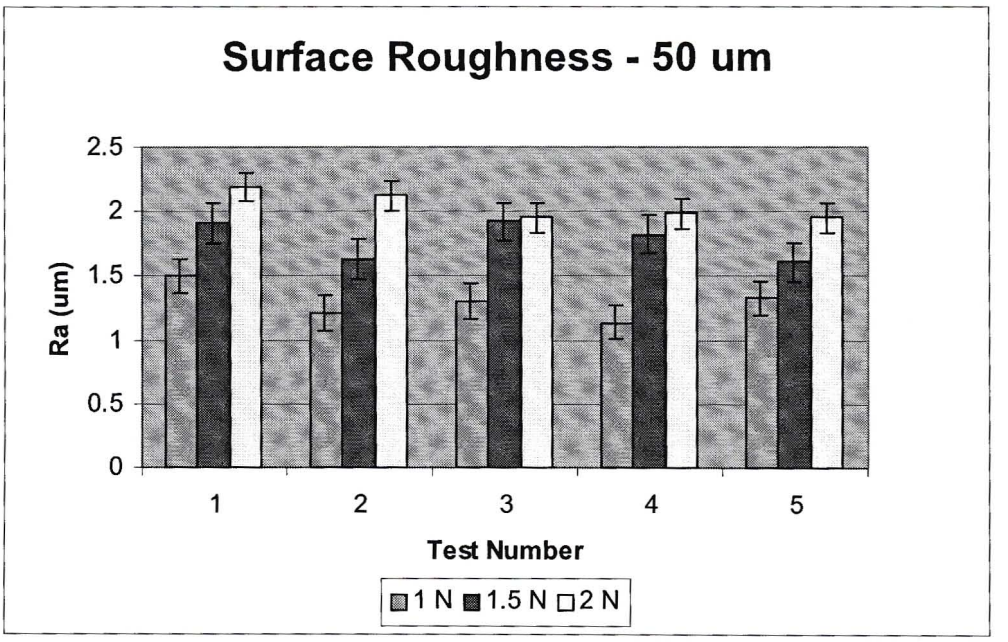
Appendix B. 7. Wear coefficient at 2 N with varying conical size

Appendix C. Surface Roughness Error

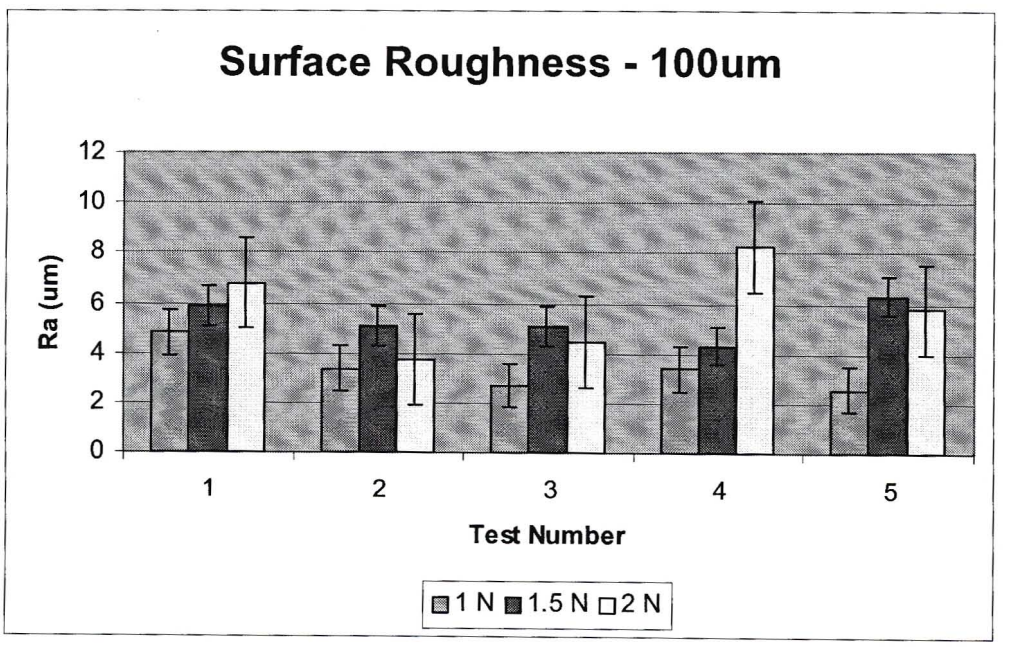
The following graphs illustrate the similarities and difference in the surface roughness values generated by the simulation. Four different mean conical sizes with three different forces were used. Each run was done five times, and the standard deviation was recorded. All simulation runs used the parameter setting from Section 5.3.1.



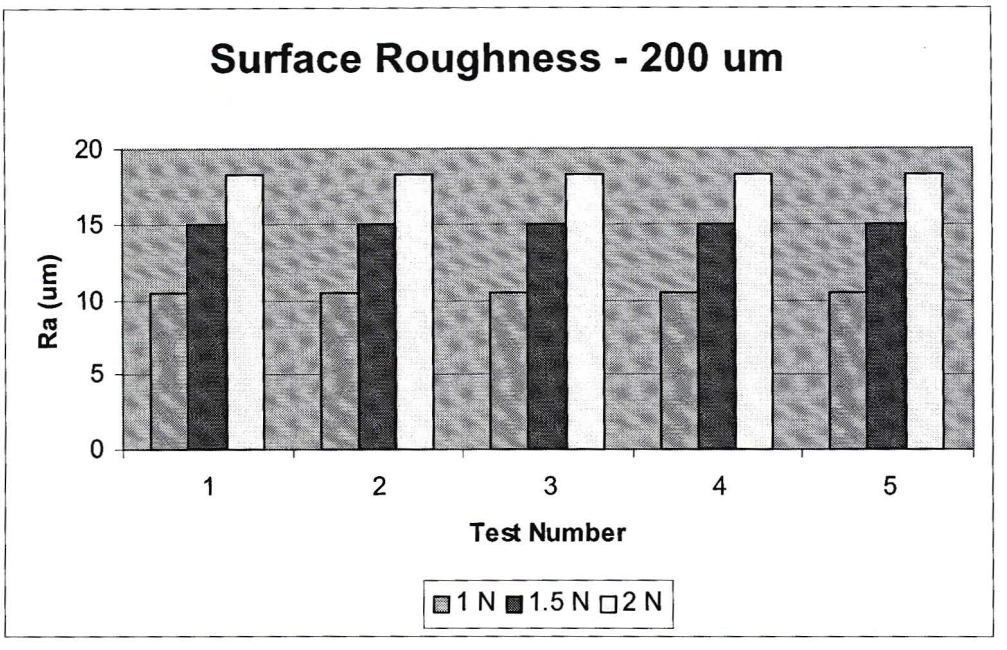
Appendix C. 1. Surface roughness at 30μm with varying force



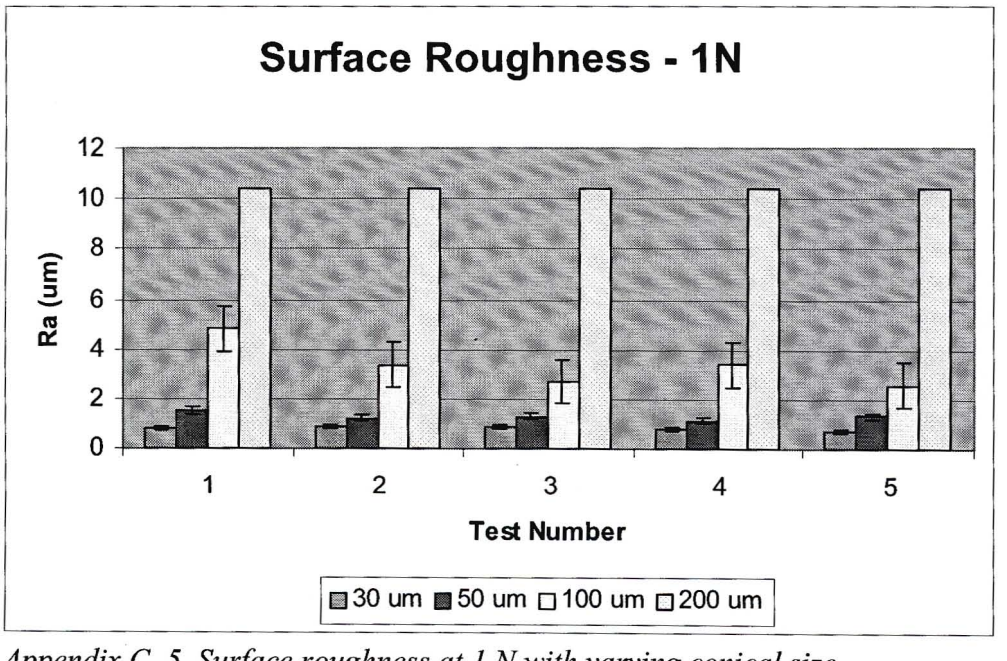
Appendix C. 2. Surface roughness at 50μm with varying force



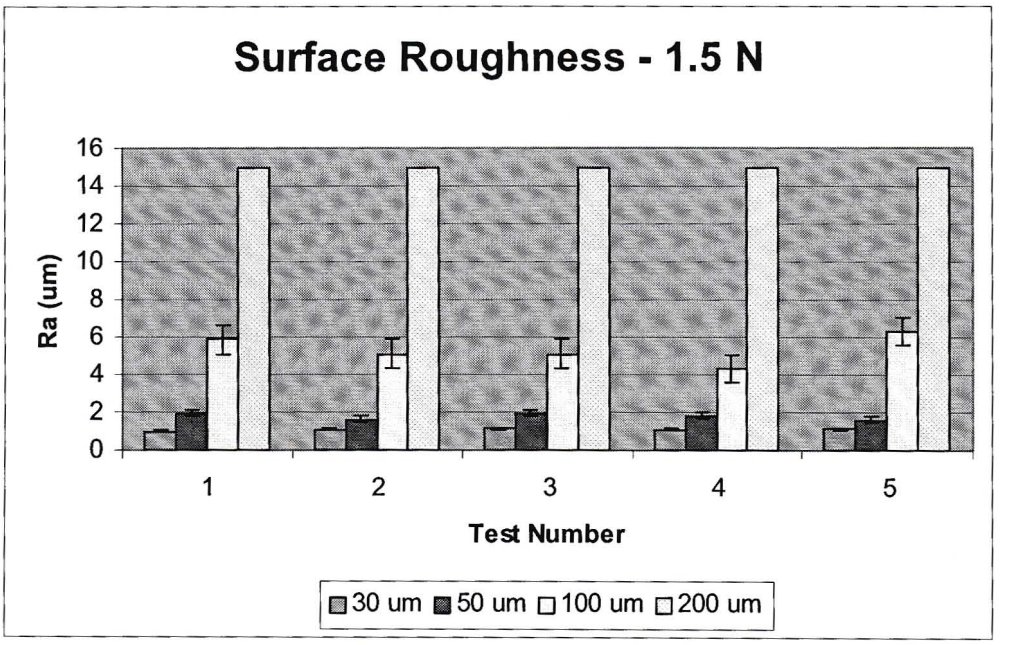
Appendix C. 3. Surface roughness at 100μm with varying force



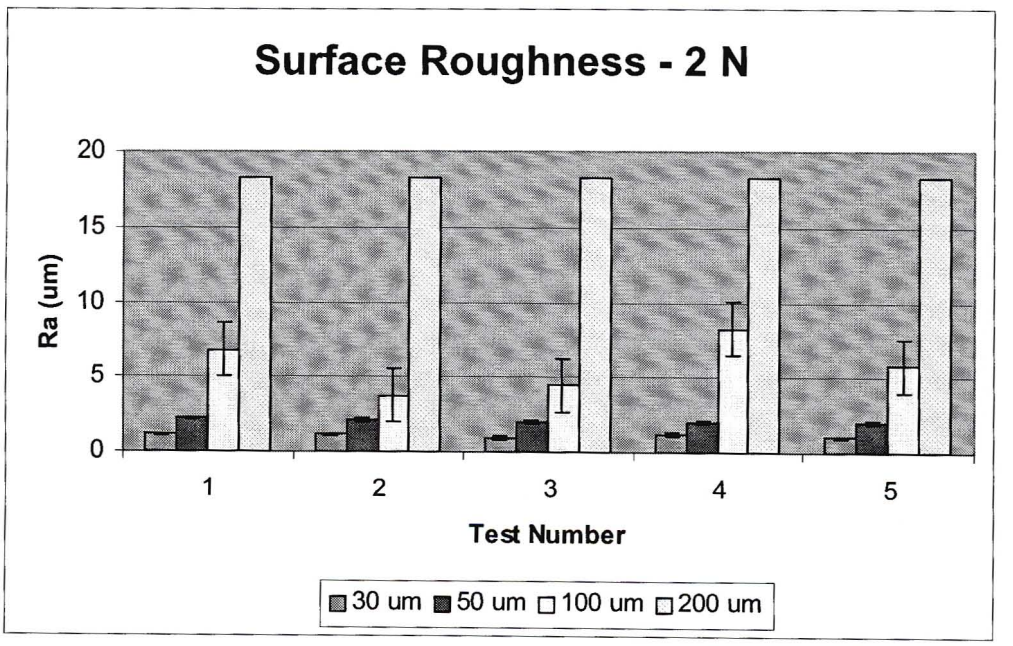
Appendix C. 4. Surface roughness at 200μm with varying force



Appendix C. 5. Surface roughness at 1 N with varying conical size



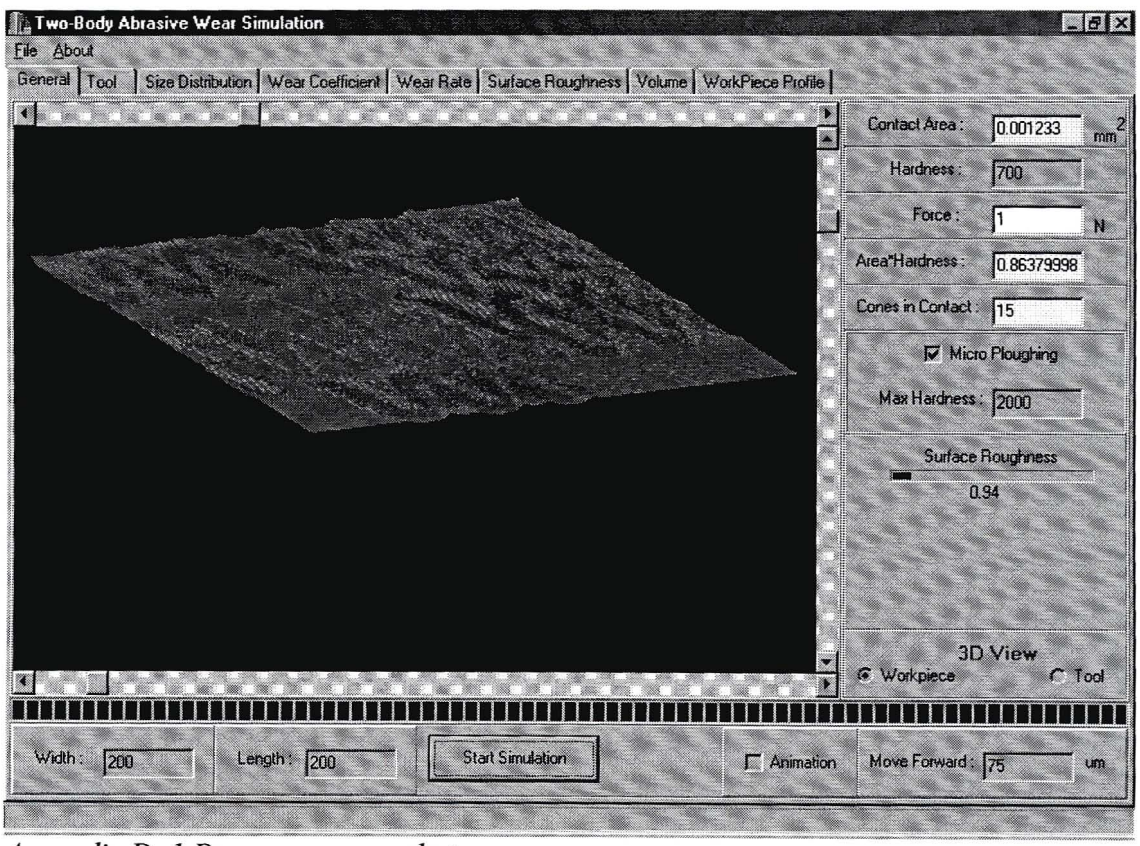
Appendix C. 6. Surface roughness at 1.5 N with varying conical size



Appendix C. 7. Surface roughness at 2 N with varying conical size

Appendix D. Program

Attached to this dissertation is a disk containing a copy of the Simulation program used in this research (there is a backup copy at http://mossie.cs.und.ac.za/~murrellh/simulation). Run the executable file found on the disk. Click the "Start Simulation" button to begin a simulation with default parameters.



Appendix D. 1 Program screen shot.

There are 8 different tabs in the program (see Appendix D. 1);

- General
- Tool
- Size Distribution
- Wear Coefficient
- Wear Rate
- Surface Roughness

- Volume
- Workpiece Profile

Each of these tabs provides data on the current TBAW simulation. Here is a simple description of each tab.

General

Provide a 3D view of the tool and workpiece. Displays the contact area between the tool and workpiece, number of contacting positions and the force and hardness used.

Tool

Shows a 2D image of the tool and enables the user to set the different parameters for the tool;

- Conical Properties - Cone Angle
- Conical Properties - Conical Tip Radius
- Tool Path - Amplitude
- Tool Path - Wavelength

Size Distribution

Displays a graph of Conical Diameter vs. Number of Conical.

Wear Coefficient

Displays a graph of Wear Coefficient vs. Sliding Distance.

Wear Rate

Displays a graph of Wear Rate vs. Sliding Distance.

Surface Roughness

Displays the change in the Workpiece Surface Roughness vs. Sliding Distance.

Volume

Displays the Volume of Material Removed from the Workpiece vs. Sliding Distance.

Workpiece Profile

Displays the cross-sectional area of the workpiece.

References

1 *Staffan Jacobson , Per Wallen and Sture Hogmark*, Fundamental Aspects of Abrasive Wear Studies by a new Numerical Simulation Model, *Wear, 123(1998) 207-223.*

2 *I. M. Hutchings,* TRIBOLOGY Friction and Wear of Engineering Materials, *Department of Materials Science and Metallurgy, University of Cambridge, Cambridge, 1992.*

3 *J. A. Williams,* Wear modelling: analytical, computational and mapping: a continuum mechanics approach, *Wear 225-229 (1999) 1-17.*

4 *K. L. Johnson,* Contact Mechanics, *Cambridge University Press, Cambridge, 1987.*

5 *Harry A. Pappo*, Simulation Model [WWW page], *URL http://home1.gte.net/simres/k1-mtype.htm, (n.d).*

6 *William K. McQuay*, Distributed Collaborative Environments for 21st Century Modelling & Simulation [WWW page], *URL http://www.modelingandsimulation.org/text/Mcquay.html, (n.d.).*

7 *Andrew P. Sage and Stephen R. Olson*, Modelling and Simulation in Systems Engineering: Whither Simulation Based Acquisitions? [WWW page], *URL http://modelingandsimulation.org/text/Sage.html, George Mason University, (n.d.).*

8 *Jiaren Jiang, Fanghui Sheng and Fengshen Ren,* Modelling of Two-body abrasive wear under multiple contact conditions, *Wear 217 (1998) 35-45.*

9 *K.-H. Zum Gahr,* Wear by hard particles, *Tribology International Vol. 31, No. 10, pp 587-596, 1998*

10 *Professor Hossein Arsham* , Systems Simulation: The Shortest Path from Learning to Applications [WWW page], *URL http://ubmail.ubalt.edu/~harsham/simulation/sim.htm, 1995.*

11 *Paul A. Fishwick*, Computer Simulation: Growth Through Extension [WWW page], *URL http://www.cis.ufl.edu/~fishwick/paper/subsection3_3_1.html, 1994.*

12 *Waite group Press*, OpenGL SuperBible, *Macmillan Computer Publishing, 1996.*

13 *Paul A. Fishwick*, Simulation Model Design and Execution [WWW page], *URL http://www.cise.ufl.edu/~fishwick/book/book.html, January 1995*

14 *Deitele & Deitel,* C++ How To Program, *Second Edition, Prentice Hall, New Jersy, 1998.*

15 *Bain Engelhardt,* Introduction To Probability and Mathematical Statistics, *Second Edition, PWS-KENT Publishing Company, USA, 1992*

16 *Sparknotes.com*, Computer Science [online study guide], *URL http://www.sparknotes.com, 1999.*

17 *Robert Ellis and Denny Gulick*, Calculus with Analytic Geometry, *Fifth Edition, Harcourt Brace & Company, USA, 1994.*

18 H. Sin, N. Saka and N. P. Sup, *Abrasive wear mechanism and the grit size effect,* Wear, 55 (1979) 163-190.

19 *Production Modeling Corporation Ltd*, About Simulations [WWW page], *URL http://fp.pmcorp.f9.co.uk, (n.d.).*

20 *E. Rabinowicz*, Friction and Wear of Materials, *Wiley, New York, 1965.*

21 *K.-H. Zum Gahr*, Modelling of two-body abrasive wear, *Wear 124 (1988) 87-103.*

22 *K.-H. Zum Gahr*, Formation of wear debris by the abrasion of ductile metals, *Wear 74 (1981 - 1982) 353-373.*

23 *K.-H. Zum Gahr*, Wear by hard particles, *in : I.M. Hutchings (Ed.), New Directions in Tribology, Mechanical Engineering Publications for the Institution of Mechanical Engineers, London, 1997, pp. 482 – 494.*

24 *S. Jacobson, P. Wallen, S. Hogmark*, Fundamental aspects abrasive wear studied by a new numerical simulation model, *Wear 113 (1988) 207-223.*

25 *S. Jacobson, P. Wallen, S. Hogmark*, Correlation between groove size, wear rate and topography of abraded surfaces, *Wear 115 (1987) 83-93.*

26 *T.C. Buttery and J. F. Archard*, Grinding and abrasive wear, *Ptoc., Inst. Mech. Eng. London, 185 (1971) 537-551.*

27 *M. J. Murray, P. J. Mutton and J. D. Watson*, Abrasive wear mechanism in steels, *J. Lubr, 104 (1982) 9-16.*

28 *Laurie Hall*, Astro-Tel, Mirror Making and other Activities by Laurie Hall [WWW page], *URL* *http://www.turbofast.com.au/astrotel/mirrorgrinding.html, Australia, (n.d.).*

29 J. A. Williams, *Analytical models of scratch hardness,* Tribology International Vol. 29, No. 8, pp. 675-695, 1996

30 Priit Põdra, Sören Andersson, *Wear simulation with the Winkler surface model,* Wear 207 (1997) 79-85

31 Y.-M. Chen, L.K Ives, J.W. Dally, *Numerical simulation of sliding contact over a half-plane,* Wear 185 (1995) 83-91

List of Symbols

a	Cone radius
A	Amplitude
A_c	Load carrying area
A_{den}	Molecular density of the abrasive material
A_{mass}	Total mass of abrasive particles
A_{vol}	Total volume of abrasive particles
D	Conical diameter size
F	Applied normal force on the tool.
$g(x)$	standard normal probability density function
h	Cone height
H	Workpiece hardness
H_a	Abrasive particle hardness
HS	Horizontal ploughing state
H_s	Workpiece surface hardness
k	Dimensional wear coefficient
K	Dimensionless wear coefficient
p	Penetration depth
Q	Wear Rate
r	Sphere radius
R_a	Surface roughness measurement in μm
s	Tools sliding distance
V	Volume of material displaced from the workpiece
V_c	Cone volume
V_s	Sphere volume
VS	Vertical ploughing state
V_t	Torus volume
w	Wavelength

x Cartesian coordinate

y Cartesian coordinate

z Cartesian coordinate