UNIVERSITY OF CAPE TOWN

Department of Mechanical Engineering

RONDEBOSCH, CAPE TOWN
SOUTH AFRICA

PROJECT No:
31

FINAL REPORT 2020

Development of an Intelligent Grinding System

Quintin Oliver de Jongh, DJNQUI001

Project Brief

The need to compete technologically in modern manufacturing has increased tremendously in machining processes. Products must be produced with high accuracy and acceptable surface integrity and increased productivity. To achieve this demands the move towards more use of machine intelligence in system is a pre-requisite. Artificial Intelligence and Machine learning have kindled the fourth industrial revolution. Fitting in this new Surface grinder with Machine Guard Present Surface Grinder 22 | Page knowledge into manufacturing systems along with data and predictive analytics will curtail raw materials, advance efficiency and optimize supply chains. The application of artificial intelligence in using computers and controllers is proven a right step forward in seeking to produce higher quality components and efficient production. Considering the use of advanced materials, grinding processes become acceptable means in the processing of components. currently, it appears to be the only practical and economical way of achieving fine surface finish, acceptable surface integrity and good geometric accuracy. Despite this importance and acceptance, grinding remains as a difficult to control process. The grinding process differs from other machining processes in complexity in the parameters and process performance because a lot of factors which are not linear are involved, as a result many processes are undertaken at a condition lower than optimal. This variable can be categorized into four major area; (a) grinding wheel, (b) workpiece material (c)operating parameters and (d) machine tools. To gain insight into grinding complexity, many researchers focused on the modelling of grinding processes. Looking into these models, some are entirely theorical, while others are practical with theoretical explanations. These research models seek to find the relationship between the process variables, wok piece quality and process parameters. Michael N. Morgan, R. C., Andrea Guidotti ,David R. Allanson, J.L. Moruzzi and W. Brain Rowe (2007). Design and implementation of an intelligent grinding assistant system Int. J. Abrasive Technology, Vol. 1, No. 1, 106-135. This is a group project for two students. One student would be working on design and development of the GUI interface, feature extraction along with correlations and the other student is expected to work on console, modelling, prediction and experimental studies.

Special note: The AML lab forms various sub-cluster groups such as: discreate machining, Polishing process for aerospace and medical components, condition monitoring and intelligent grinding system. This particular project falls Industry 4.0. The student is expected to work with our PhD student Ms Alice Alao on this project.

SUPERVISOR: *A/Prof Ramesh Kuppuswamy*

Word Count: 16513

Declaration

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used the IEEE (The Institute of Electrical and Electronic Engineers), convention for citation and referencing. Each significant contribution to, and quotation in, this report / project from the work(s) of other people has been attributed and has been cited and referenced.
3. This report/project is my own work.
4. I have not allowed and will not allow anyone to copy my work with the intention of passing it off as his or her own work.

Signature: _____

Table of Contents

List of Figures 5

List of Tables 7

Nomenclature, Acronyms and Definitions 8

Acknowledgments 10

1 Abstract 11

2 Introduction 13

2.1 Problem Statement 13

2.2 Project Impact 13

2.3 Objective and Scope 14

2.4 Project Organisation 14

3 Literature Review 16

3.1 Background to the Grinding Process (namely Surface Grinding) 16

3.2 General Intelligent Grinding (and Machining) Systems 19

3.3 Materials and Ti-6Al-4V in Particular 24

3.4 Feature Extraction & Selection 27

3.5 Modelling 28

3.5.1 Numerical Modelling 28

3.5.2 Experimental/Empirical Modelling 28

3.5.3 Micro-Analytical Modelling 29

3.5.4 Macro-Analytical Modelling 31

3.6 Residual Stress, Temperature and Coefficient B 32

3.7 Signal Processing & Noise Elimination 34

3.8 Wheel Wear/Life and Kurtosis Coefficient 35

3.9 Database 37

3.10 Human-Machine Interface 37

3.11 Feature Correlation & Machine Intelligence 39

3.12 Summary of Literature Review 41

4 Methodology 42

4.1 Feature Extraction 42

4.1.1 Analytical/Semi-Empirical Modelling 42

4.1.2 Final Empirical Model Used in the System 51

4.1.3 Experimental Mock Data Creation & Feature Correlation 56

4.1.4 Past Experimental Data Use 62

4.1.5 Noise Elimination and Statistical Analysis 64

4.2 Database 66
4.3 Intelligent System Development and Human Machine Interfaces (Graphical User Interfaces) 69
5 Results and Discussion 76
5.1 Full Integration of the System 76
5.2 Developed IGS Package & Console 79
5.3 MATLAB Model Developed Results and Discussion 80
5.4 Full Working Example of the System 89
6 Conclusions 99
7 Recommendations 102
8 Reference List 104
9 Appendices 108
9.1 MATLab Model Code 108
9.2 MATLAB Code for Noise Elimination 112
9.3 MATLAB System Database Updates, Retrievals, Inserts and Deletes 114
9.4 MATLAB System App Code (Body-Only) 118
Risk Assessment Form 132

List of Figures

Figure 1 - Macro-Grinding Diagram [43] 17
Figure 2 - Enlarged Grit View of Surface Grinding [2] 17
Figure 3 - Abrasive Grain Trajectory and Surface Created [2] 18
Figure 4 - Amitay Control System for Grinding [5] 20
Figure 5 - Bhusan, Acquirable Parameters [28] 20
Figure 6 - GIGAS Optimization Process [8] 21
Figure 7 - Li Process Flow Chart [11] 22
Figure 8 - Morgan et al. Schematic Diagram [4] 23
Figure 9 - Ti-6Al-4V Material Varying Properties [46] 24
Figure 10 - Phase Transformation Ti-6Al-4V [44] 25
Figure 11 - Phase Transformation Ti-6Al-4V, quenching [7] 26
Figure 12 - Flowchart for Grit Modelling (Micro) [24] 30
Figure 13 - Alloy Steel - Residual Stress vs Coeff. B [30] 33
Figure 14 - Bearing Steel - Residual Stress vs. Coeff. B [30] 33
Figure 15 - LP and HP Filters [31] 34
Figure 16 - AE Signal Denoising [31] 34
Figure 17 - Denoising Logic [31] 35
Figure 18 - Abrasive Wear, Tribology [47] 35
Figure 19 - General Forms of Kurtosis [45] 36
Figure 20 - Rowe et al. GUI [35] 38
Figure 21 - Morgan et al. GUI [4] 38
Figure 22 - Choi and Shin Interfaces [8] 39
Figure 23 - Kuppuswamy & Airey - Neural Network [21] 40
Figure 24 - Initial Model Flowchart 43
Figure 25 - Initial Model Flowchart (Formulae and Parameters) 44
Figure 26 - Final Model Flowchart 52
Figure 27 - Force vs. Depth of Cut Vw = 10000mm/min, Vs = 30m/s 57
Figure 28 - Surface Plot Vw = 10000mm/min, Vs = 30m/s 58
Figure 29 - Surface plot 2, Vw = 10000mm/min, Vs = 30m/s 58
Figure 30 - Coefficient B vs. Residual Stress Vw = 10000mm/min, Vs = 30m/s 60
Figure 31 - Residual Stress vs. Depth of Cut - Vw = 10000mm/min, Vs = 30m/s 60
Figure 32 - Grinding Normal Force vs Time for a Full Experiment (DeweSoft) 62
Figure 33 - Grinding Normal Force vs Time for Four Passes (with annotations) 62

Figure 34 - Extracted Excel Experimental Data Part 1 63
Figure 35 - Extracted Excel Experimental Data Part 2 63
Figure 36 - Noisy Force Signal (imported into MATLAB) 64
Figure 37 - Corresponding Denoised Signal 64
Figure 38 - Database Structure 66
Figure 39 - Material Table 68
Figure 40 - Reference Cycle Table 68
Figure 41 - All Tables, Queries and Forms of Database 68
Figure 42 - IPR Table 68
Figure 43 - Database GUI Main Menu 69
Figure 44 - Material Table GUI 70
Figure 45 - Output and Recommendations App Design 71
Figure 46 - Output Page Portion of Methods Code 72
Figure 47 - Methods Code Portion 2 73
Figure 48 - Methods Code Portion 3 73
Figure 49 - DeweSoft Channels 74
Figure 50 - DeweSoft Export Instructions 74
Figure 51 - DeweSoft Run page GUI 75
Figure 52 - Full System Integration Flowchart 76
Figure 53 - Console and Package 79
Figure 54 - Normal Force vs Depth of Cut - Vw = 10000mm/min, Vs = 60m/s 80
Figure 55 - Residual Stress vs. Depth of Cut - Vw = 10000mm/min, Vs = 60m/s 81
Figure 56 – Coefficient B vs. Residual Stress - Vw = 10000mm/min, Vs = 60m/s 81
Figure 57 - Normal Force vs Depth of Cut - Vw = 12500mm/min, Vs = 45m/s 83
Figure 58 – Coefficient B vs. Residual Stress - Vw = 12500mm/min, Vs = 45m/s 84
Figure 59 - Residual Stress vs. Depth of Cut - Vw = 12500mm/min, Vs = 45m/s 84
Figure 60 - Normal Force vs Depth of Cut - Vw = 15000mm/min, Vs = 30m/s 86
Figure 61 - Coefficient B vs Residual Stress, Vw = 15000mm/min, Vs = 30m/s 87
Figure 62 - Residual Stress vs. Depth of Cut - Vw = 15000mm/min, Vs = 30m/s 87
Figure 63 - IGS Main Menu (MATLAB) 89
Figure 64 - Guide to Use Page (MATLAB) 90
Figure 65 - IGS Start Page (MATLAB) 91
Figure 66 - IGS Wheel, Work, Coolant Input Page (MATLAB) 92
Figure 67 - Material, Wheel, Coolant Tables (Access Database) 93

Figure 68 - IGS Grinding Parameters Input (MATLAB) 93
Figure 69 - Updated Active Cycle (Access Database) 94
Figure 70 - IGS Outputs and Recommendations, Predicted Outputs (MATLAB) 94
Figure 71 - IGS Outputs and Recommendations, Calculated Values (MATLAB) 95
Figure 72 - IGS Run-page (DeweSoft) 96
Figure 73 - IGS Homepage (DeweSoft) 96
Figure 74 - Updated Reference Cycle Table (Access Database) 97
Figure 75 - Updated IPR (Access Database) 97
Figure 76 - IGS Analysis Page (MATLAB) 98
Figure 77 - Updated Optimised Table (Access Database) 98

List of Tables

Table 1 - Results of Model Vw = 10000mm/min, Vs = 30m/s 59
Table 2 - Results Summary - Vw = 10000mm/min, Vs = 60m/s 82
Table 3 - Results Summary - Vw = 12500, Vs = 45m/s 85
Table 4 - Results Summary - Vw = 15000mm/min, Vs = 30m/s 88

Nomenclature, Acronyms and Definitions

Industry 4.0 – The integration and development of automation and intelligence in manufacturing technologies

MATLAB – Matrix Laboratory, a matrix-based programming language and numerical computing environment developed by MathWorks

DeweSoft – A supplier of data acquisition systems (both hardware and software), focused on ease of use

Machine Intelligence – The use of AI in manufacturing systems for machines to interact with certain environments in an intelligent manner

AI – Artificial Intelligence

Ti-6Al-4V – Titanium Alloy, containing 6% Aluminium and 4% Vanadium

NN – Neural Network

Asperity – Uneven surfaces (roughness), pertaining to high spots on the surface of materials

Macroscopic – Large-scale/general analysis that is visible to the naked eye, involving general elements of systems

Microscopic – Small-scale analysis pertaining to investigation of individual elements in a broader system

Ft – Tangential grinding force (cutting force)

Fn – Normal grinding force (perpendicular to cutting)

ACC – Adaptive Control with Constraints

ACO – Adaptive Control with Optimization

LVDT – Linear Variable Differential Transformer

DAQ – Data Acquisition

GUI – Graphical User Interface

GIGAS – Generalized Intelligent Grinding Assistant System

CBR – Case Based Reasoning

RBR – Rule Based Reasoning

CNC – Computer Numerical Control

IGA – Intelligent Grinding Advisory System

HCP – Hexagonal Closed Packed Crystal Structure (HCC), alpha phase

BCC – Body-Centred Cubic Packed Structure (BCP), beta phase

AE – Acoustic Emissions

DWT – Discrete Wavelet Transformation

PCD – Polycrystalline Diamond

FEM – Finite Element Methods

LP – Low Pass

HP – High Pass

BHN – Brinell Hardness

IGS – Intelligent Grinding System

Vw – Workpiece speed (feed-rate)

Vs – Wheel speed

UCT – University of Cape Town

IPR – Individual Part Record

MatWeb – Online Materials Information Resource

Acknowledgments

I would like to express my grateful appreciation to Associate Professor Ramesh Kuppuswamy for his continued support, valuable knowledge imparted upon me, useful critiques, and incredible guidance through the duration of the project. He has kept me focused and motivated throughout the year and helped me find a passion in the field of mechanical engineering.

I would also like to extend my appreciation to Mr. Fungai Jani, who supported my research and assisted the development of my project in a patient yet focussed manner.

I would like to say thank you to those closest to me, who have gifted me with emotional support when I needed it most.

Finally, I wish to express my appreciation, that cannot be put into words, to my mother, for all she has done for me.

1 Abstract

Industry 4.0 is growing at a rapid pace and the development of technology is expected to keep up with it. In addition to this, the modern manufacturing industry is very competitive due to the high expectations with regards to surface finish, speed of processing, dimensional accuracy and low wheel wear. A noticeable increase in demand for machine intelligence and implementation of prediction models in machining systems has been observed since the development of modern computing systems [4].

This report describes the creation and continued development of an intelligent grinding system, which employs a semi-empirical-analytical model at its core, to allow for prediction and analysis of grinding conditions based on workpiece material, wheel type, and grinding process parameters. The outputs of the system are namely: workpiece surface temperature, coefficient B (to be described later) and residual stress.

'Mock' experimental data has been created using a semi-empirical-analytical model, which was presented to the other team member on this project (Kalvin Govender), who was developing a feature correlation engine, namely with the use of a neural network. A few past experiments were analysed as well (from a previous student) and the model was run on it to provide more data to the other student on the team.

An intelligent database was developed to support the system and to allow for storage and referencing of data, as well as providing the basis for optimisation of the process. The database contains tables relevant to storage of important material, wheel, and coolant information, and stores the inputs by system operators as well as the corresponding outputs of the system itself.

Three graphical user interfaces have been developed. A database GUI has been created to allow for addition, viewing and deletion of data. A system user interface was created that accepts inputs from operators and displays predictions and outputs to this operator as well. A run-time interface was created to allow for live running of the grinding process (theoretical in nature though). These three interfaces were developed with Microsoft Access Forms, MATLAB App Designer and DeweSoft. The backend of each system (database, MATLAB algorithms and DeweSoft data collection/exports) communicate through the developed MATLAB program/app, which is the heart of the created system.

Live acquired signals through the DeweSoft system are exported for analysis in MATLAB, and these signals are eliminated of noise through the combined use of discrete wavelet transformations and

moving average filters. This allows for the calculation of the kurtosis coefficient, which gives an indication of wheel wear.

Due to apparent and extenuating circumstances that have become evident this year, namely the COVID-19 pandemic, many resources were not available, and thus the scope was moulded appropriately from what it had initially been. Experimental studies were no longer an option as access to laboratories were prohibited, thus, a theoretical modelling approach, mixed with past experimental studies and prediction modelling, was used to complete the project.

2 Introduction

2.1 Problem Statement

Grinding is an economical means to achieving required surface properties with high geometrical accuracy. Shorter cycles times with less operator interventions are desirable in the process and it is known that optimization of the process results in increased productivity and component quality, while maintaining lower cost. Experience of operators has been an important aspect to consider in the grinding process, and when relying on this experience, suboptimal conditions can result.

The above requirements lead to the use of newer technology, that of machine intelligence. This involves applying AI techniques to in-process models to create a system that runs more efficiently and in the most optimal manner possible, learning from itself after every iteration.

Many predictive models and intelligent grinding systems have been developed in the past; however, they lack the live analytical and predictive abilities that are required for industry. The real-time feature allows for preventative measures to be implemented to eliminate failure as well as to suitably modify grinding parameters while the process is taking place.

A system with built-in intelligence and adaptability to many grinding conditions, with the ability to effectively predict and monitor the grinding process, must be developed in order to solve this problem.

The project aims are: develop a surface grinding model that can be modified based on workpiece material and grinding wheel, create a structured database for effective storage, collection and retrieval of data, and create human-machine interfaces that allows for ease of operation with minimal experience and learning time required. Subsequent aims are: develop usable 'mock' experimental data for feature correlation, denoise force signals effectively, and to investigate and report on the material science behind the TI-6Al-4V alloy.

The broader project aim is to develop a system capable of grinding condition state prediction (namely workpiece surface temperature and residual stress magnitude) as well as real-time conditional monitoring.

2.2 Project Impact

The research and developed technology will aid in prediction of grinding conditions and will help in making the process more efficient, less wasteful and easier to implement by operators of all skill levels. The scrap rate of materials and wear rate of grinding wheels will be reduced, and the grinding process shall be easier to control. The developed system will be of interest to both institutions/universities, and to industry, as it can help with both manufacturing and experiments.

The research regarding Ti-6Al-4V will be of interest to industries such as those of the aerospace, automotive and biomedical, as this material is very apparent in these industries.

2.3 Objective and Scope

The objective of this project is to develop an intelligent grinding system that can predict grinding conditions by focussing on the limitations of workpiece burn and residual stress formation in the workpiece. The project was divided into a group of two and the scope has been clearly defined for each member so that no intersection of scope occurred, and so that work could occur simultaneously without the need for the other member's work initially. The scope of this project was limited (and further extended) to the feature extraction engine (development of the grinding model and selection of important features), database development and creation, and human-machine interface development. The other student's scope was to develop a feature correlation engine (machine learning via neural networks) and optimisation/recommendations of conditions to be used. The other student's work could not be implemented into this system and so necessary adjustments have been made (as is seen throughout the report) and comments have been added throughout the report to show how implementation would occur.

The scope of this project also includes the conduction of a literature review, analysis of results, evaluation of performance and to draw conclusions and make recommendations on the project.

2.4 Project Organisation

This report is organised in a manner such that it reveals all necessary information required initially to understand the project, before describing the more detailed sections and results. Organization of the report follows a rough timeline from the beginning of the project to the end of the project. In order, the following chapters are included:

- The report begins with an abstract, which describes the topic, approach to the project, and main outcomes and conclusions of the project, in a succinct manner.
- The introduction is then presented and provides a background to the report (with context) as well as a clear statement of purpose, technical background and impact of the project. Organisation of the report is included too, of course.
- A literature review is then presented which shows research relevant to building the basis of the project. Concepts that were deliberated and work by other researchers is included here too.

- The methodology of project creation is then submitted, and it contains the design of the grinding system, namely: feature extraction engine (modelling), database and human-machine interfaces.
- The results and discussion chapter follows, whereby a full description of the integration of all system components is discussed, followed by a discussion of a proposed complete hardware-to-software package and then a discussion and presentation of results from the model and finally a presentation of a full working example of the system.
- A conclusion of the report follows, where a summary of the findings is presented along with a highlight of the major outcomes.
- Recommendations on how the project could be further developed or improved on, are then made.
- Finally, multiple appendices are provided, which show all the code developed in MATLAB, the results developed from the model, along with other supporting code and results. A risk assessment form is included in the appendix to fulfil the required exit level outcome while the marked interim report with all necessary forms, has been uploaded to the Vula site drive, as instructed. No expenditure was used on this project and thus there was no budget required and this could be omitted from the report.

3 Literature Review

3.1 Background to the Grinding Process (namely Surface Grinding)

Grinding is an abrasive, finishing process used to remove material and it is instrumental in providing required surface properties [1]. The process is regarded as having a low efficiency because the specific energy required to grind is much greater than that of other machining and cutting processes [1]. This inefficiency, however, often allows required material properties of a workpiece to be obtained in a single pass [1].

The process of material removal in grinding is most importantly understood when looking at the interaction of the abrasive wheel's grain geometry with the surface topography of the workpiece [2]. The wheel's abrasive grains should be significantly harder than that of the workpiece material. When the asperities of the wheel's grains are forced into the surface of the workpiece, a small amount of material is removed (cut away). It is important to note that in the grinding process, the grains are not well defined with regard to orientation and geometry. Grains are also fixed or bonded in the grinding process (as opposed to loose abrasive processes such as polishing or lapping) which means that particles are held together in a matrix and the combination of their geometries determine final workpiece geometry.

Grinding is mainly used to machine hard materials and to achieve high accuracy with regards to size, shape and surface finish [3]. The particularly hard abrasives (known as super abrasives) used in the matrix of grain abrasives are well suited to challenges presented by newer engineering materials such as that of the hard, yet brittle, ceramic material [3].

Grinding force is a particularly important parameter and again, aids in understanding the process. Characteristic to grinding is that of the high ratio of normal force to tangential force, which results in high temperatures and specific energies [1]. Grinding energy can be looked at from a macroscopic or microscopic lens, where the former describes the energy used for the entire operation while the latter describes the energy used during the cutting action of a single grit [12]. What is important to take away from this is that the specific grinding energy can be found from either method and this describes the energy required to remove one volumetric unit of material [12]. The high temperature associated with grinding can be very detrimental to workpiece quality as two notably negative effects can occur to the workpiece: surface burns and/or residual stress accumulation. If a workpiece burns or acquires too much residual stress, it cannot be used and must be scrapped. However, if these effects go unnoticed and the workpiece is used anyway, catastrophic failure can occur.

Figure 1 below shows a schematic diagram of the surface grinding procedure with cutting force (Ft), normal force (Fn), wheel speed (Vc), workpiece speed (Vw) and depth of cut (a) labelled appropriately. Abrasive grains are illustrated on the grinding wheel as well.

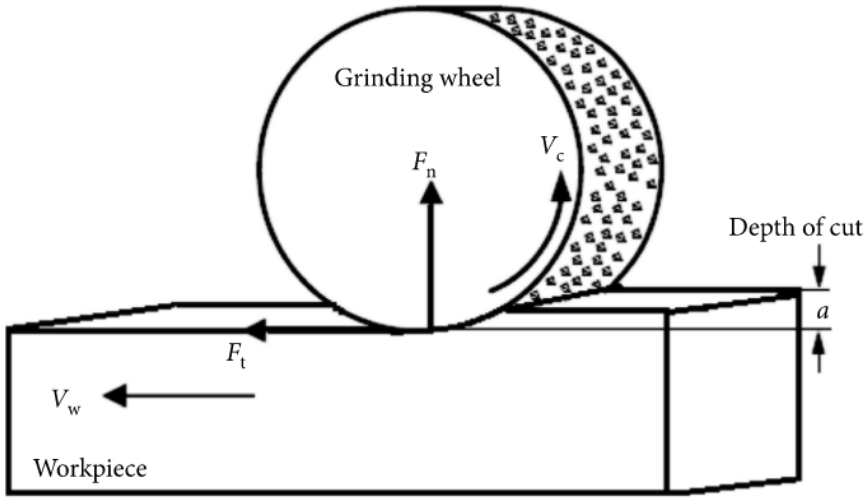

Figure 1 - Macro-Grinding Diagram [43]

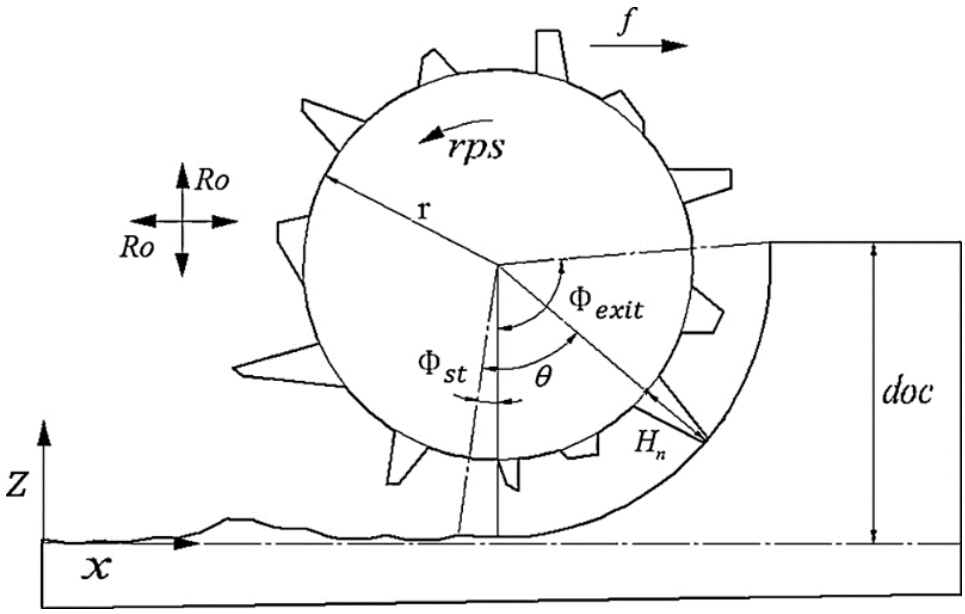

Figure 2 - Enlarged Grit View of Surface Grinding [2]

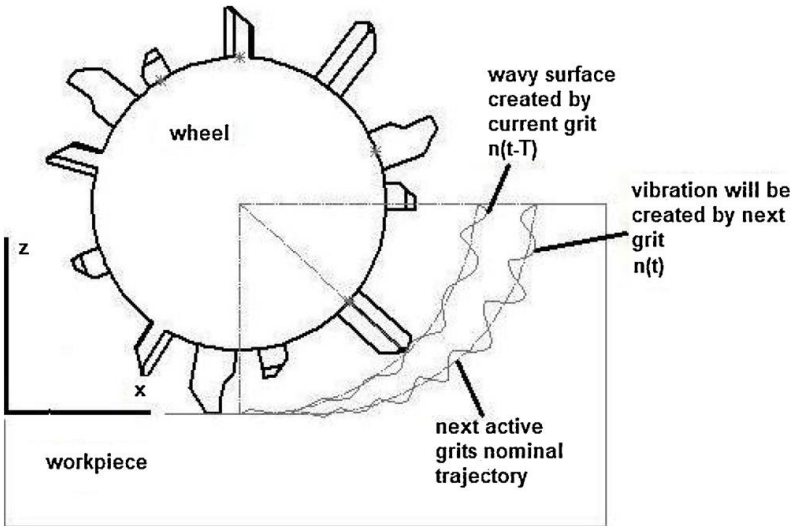

Figure 3 - Abrasive Grain Trajectory and Surface Created [2]

Figure 2 (on the previous page) and Figure 3 (above) show a better view of wheel abrasive grains interacting with the surface of a workpiece. This manner of approaching grinding is particularly important when modelling the process in a mathematical/analytical sense whereby the physics-based nature of the grinding process is investigated.

3.2 General Intelligent Grinding (and Machining) Systems

Optimization of systems is one of the most important ways in which we achieve efficiency and consistent quality. Up to date, productivity and efficiency of the grinding process has often relied on operator experience [4]. As a result, suboptimal conditions were often in place when grinding and this in turn produced suboptimal results, where turnover was lower than ideal and scrap rates were higher than necessary. The emergence and rapid evolution of newer technology, in this case: easier monitoring and analysis of parameters using computers, machine/artificial intelligence, simulation, modelling, data storage, statistical analysis and interface development, have allowed for the development of intelligent grinding systems, where artificial intelligence techniques can be applied to in-process models and machine selection techniques [4]. Efficient experimental validation is also a key benefit of modern machining, meaning that developed simulations and models can be quickly verified.

It should be noted that while intelligent grinding systems can be broad in their sphere of focus, many of the systems focus more on some aspects of optimization (sometimes only focussing on one aspect). Some examples of focus aspects are wheel wear, surface roughness of workpieces, dress conditions and geometric accuracy.

The premise of an intelligent control system is that of adaptive control, which refers to control of operating parameters in reference to process characteristic measurements to operate at desired conditions [5]. Adaptive control is further categorised into adaptive control constraint (ACC) and adaptive control optimisation (ACO) [5]. ACC has been more common in the past as it only entails specification of fixed constraints to a desired condition [5]. This makes ACC a simpler route to follow when designing a control system. ACO involves maximizing an index of performance and requires a suitable index and control policy. Developing an uncomplicated control policy and finding a suitable index has created a large difficulty in the past and thus many developers have shied away from using ACO [5].

Amitay et al. developed an intelligent control system for grinding and used optimal loci and in-process monitoring to optimize the dressing and grinding parameters for plunge-grinding of steels, maximizing removal rate with the constraints of surface finish and workpiece burn [5]. Figure 4 on the following page shows the online adaptive control system implemented by Amitay et al. [5], which incorporates an optimal locus optimization strategy at the heart of the system. Pb is burning power, e is the error, vf is the infeed velocity and nw is the spindle sped. The raw theory and description of control systems at their core as well as the theory of the optimization strategy (optimal loci) are what have been taken from Amitay et al.'s project and applied to this project.

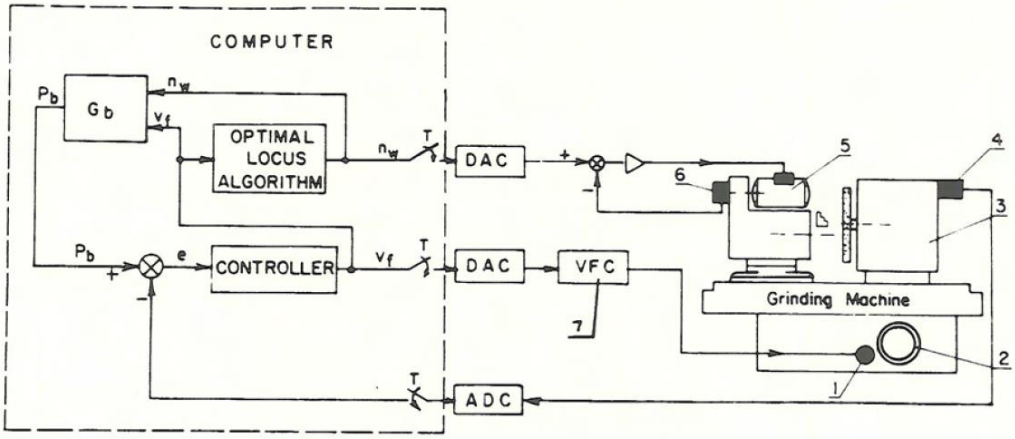

Fig. 5 On-line adaptive control system: 1-stepping motor infeed drive; 2-infeed control handwheel; 3-grinding wheel motor; 4-power sensor; 5-workpiece spindle DC motor; 6-tacho-generator; 7-voltage-to-frequency converter

Figure 4 - Amitay Control System for Grinding [5]

Bhusan [28] developed a portion of a diagnostic tool for in-process modelling of grinding by using power and displacement sensors, once again confirming a relationship between material removal rate and peak power. More sensors and the integration of the LVDT (Linear Variable Differential Transformer) with the DAQ (Data Acquisition System) would have been required to develop a more complete package but the work done in order to acquire signals was notable.

In the development of Bhusan's intelligent system, the parameters that can be measured during and after the grinding process has occurred, were stated. These can be seen in Figure 5 below. The parameters that can be measured in-process and which are chosen to be measured, determine the development of control strategies as well as inputs and outputs of the system. The in-process monitoring chosen by Bhusan was that of power monitoring as this is a simple signal to measure directly and many other grinding parameters can be derived from this parameter alone. The use of in-process monitoring of a single, easily acquirable parameter is what has been taken from Bhusan's work and applied to this project.

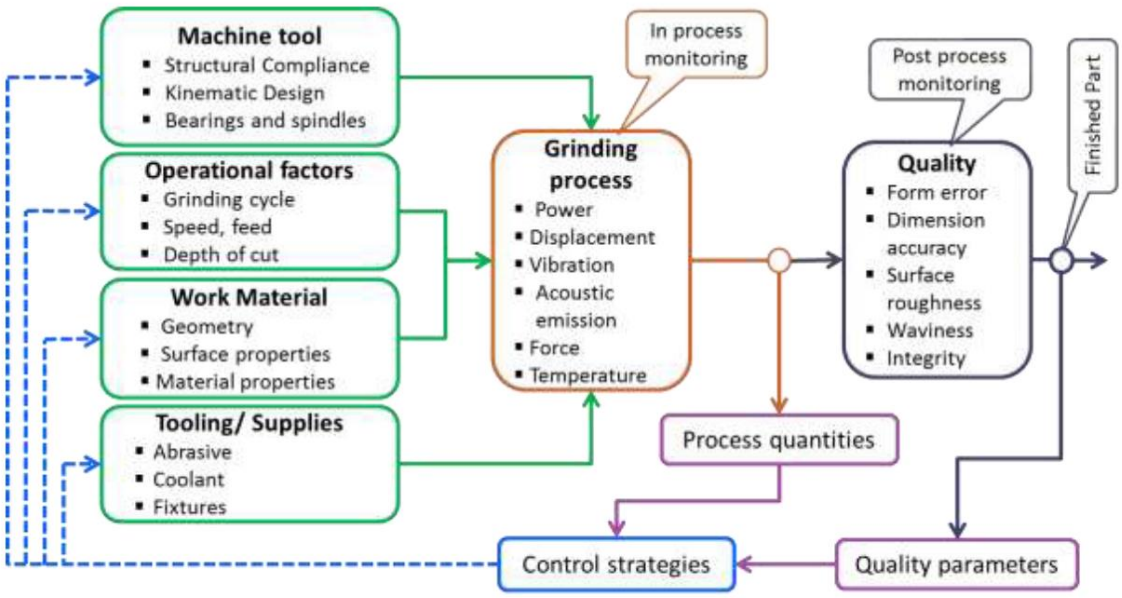

Figure 5 - Bhusan, Acquirable Parameters [28]

Choi and Shin [8] developed a system that they termed the 'generalized intelligent grinding advisory system' (GIGAS). They modelled the grinding process (and acceptable process conditions) and formulated a problem regarding input, continuous and output variables. They then structured the system by creating a model and machine database, an optimization engine and finally a graphical user interface (GUI). The system was notable in the sense that it did not compromise on process requirements. It should also be noted that the system learnt through a combination of experimental data, analytical models and heuristic rules [9]. Figure 6 below shows the optimization process created for GIGAS. The model creation, aspects of database and aspects of the optimization engine are what have been investigated here and used in this project.

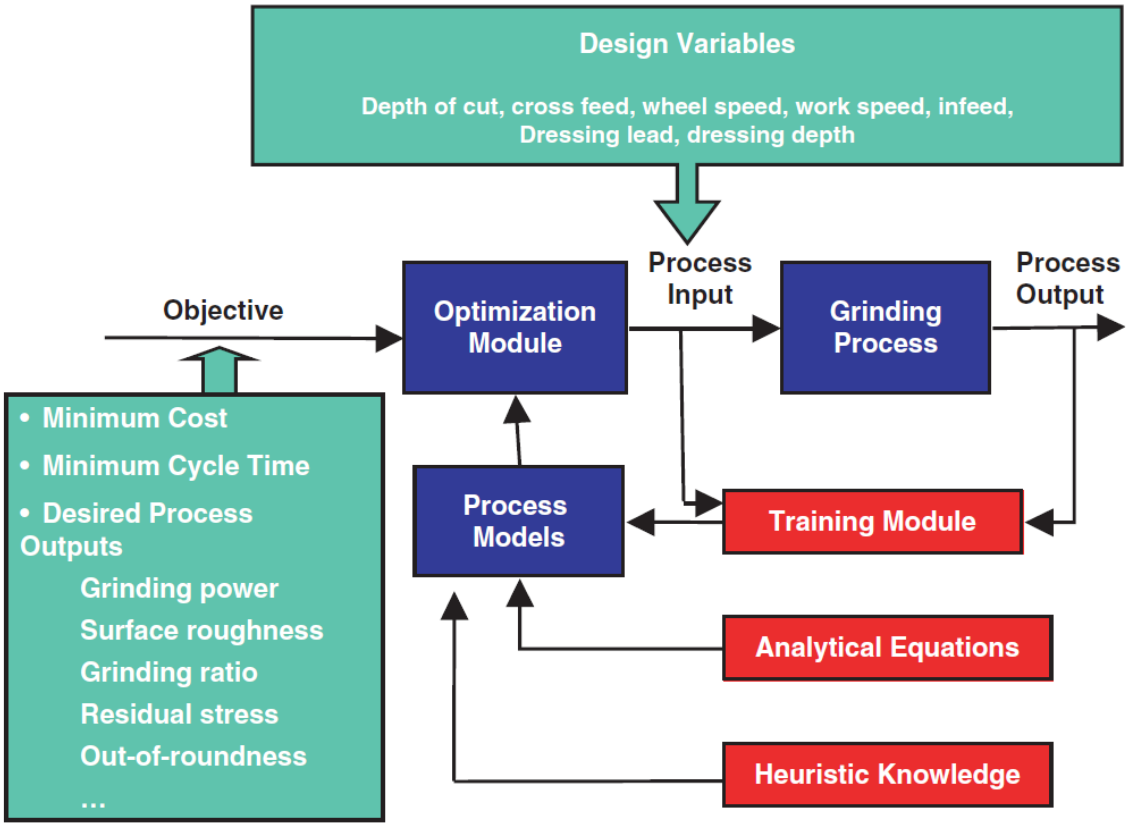

Figure 6 - GIGAS Optimization Process [8]

Li [11] developed a system for intelligent selection of grinding conditions, using a hybrid technique of CBR (case-based reasoning), NN (neural networks) and RBR (rule-based reasoning). The system would ask for inputs such as wheel type, workpiece material and process conditions and would output properties such as dressing parameters and recommended grinding wheel. Figure 7 below shows the control flowchart for the system. The system is particularly notable for its flexibility and automatic updating and retrieval of information from the database [11]. This notable feature is what has been take from Li's project and applied to this project, as well as aspects of the process flow chart shown in Figure 7 below. The multi-methods of machine intelligence allow for overcoming of weaknesses from the other machine intelligences and combine to provide a much stronger system overall, with more accurate predictive abilities and thus more suitable grinding conditions. This system was designed for external cylindrical plunge grinding [11]. Note that an adaptive control system was not incorporated into this system.

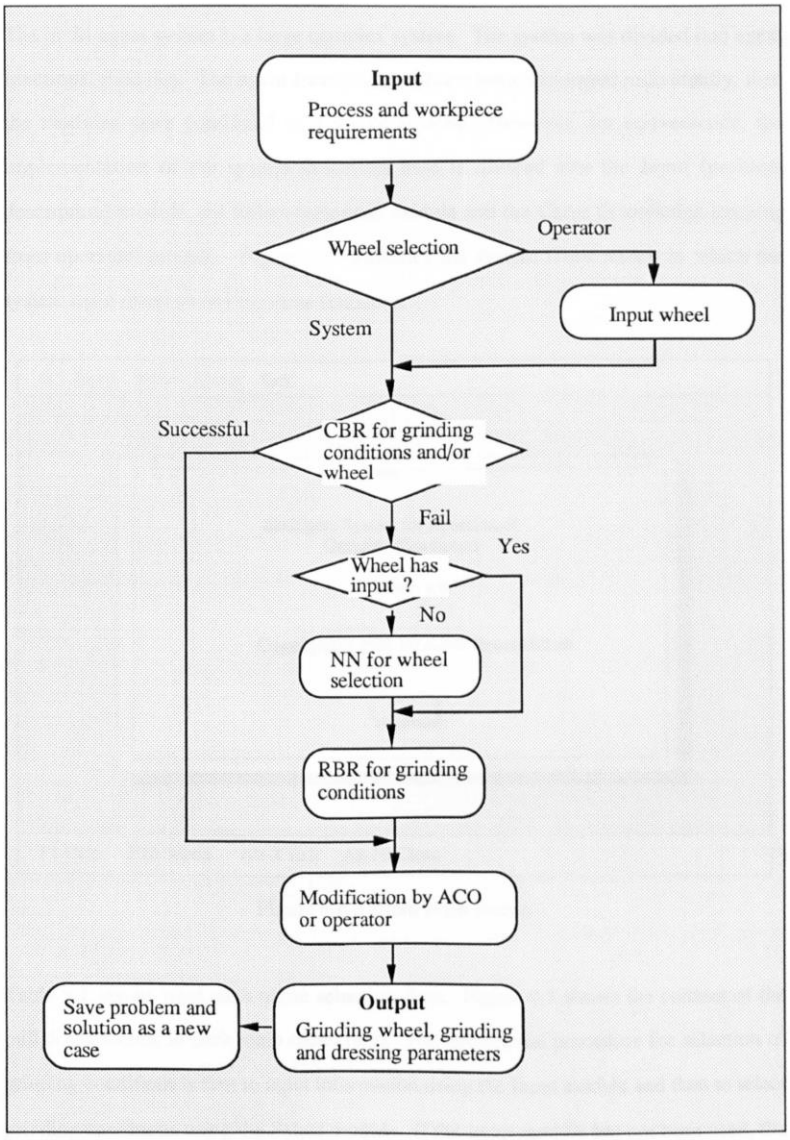

Figure 7 - Li Process Flow Chart [11]

Ending on the most notable system to date and the system of most relevance to this project, that of Morgan et al. [4]. They developed a system termed 'IGA' (intelligent grinding advisory system). It is a software package that links to the instrumentation and CNC on a grinding machine. The most notable feature is that analysis of the grinding performance of the machine is done in real time [4]. The algorithms incorporated in the software alter certain parameters in order to improve the grinding performance [4] (notably to prevent workpiece burn and ensure fastest possible cycle time). A database was created in the package to store data needed for the algorithms to occur. These include material properties of the workpiece, wheel, and coolant, respectively, as well as certain grinding parameters for each cycle. Detailed machining parameters and critical part monitoring are highlighted aspects of the system. Figure 8 below shows the schematic diagram of the complete IGA system.

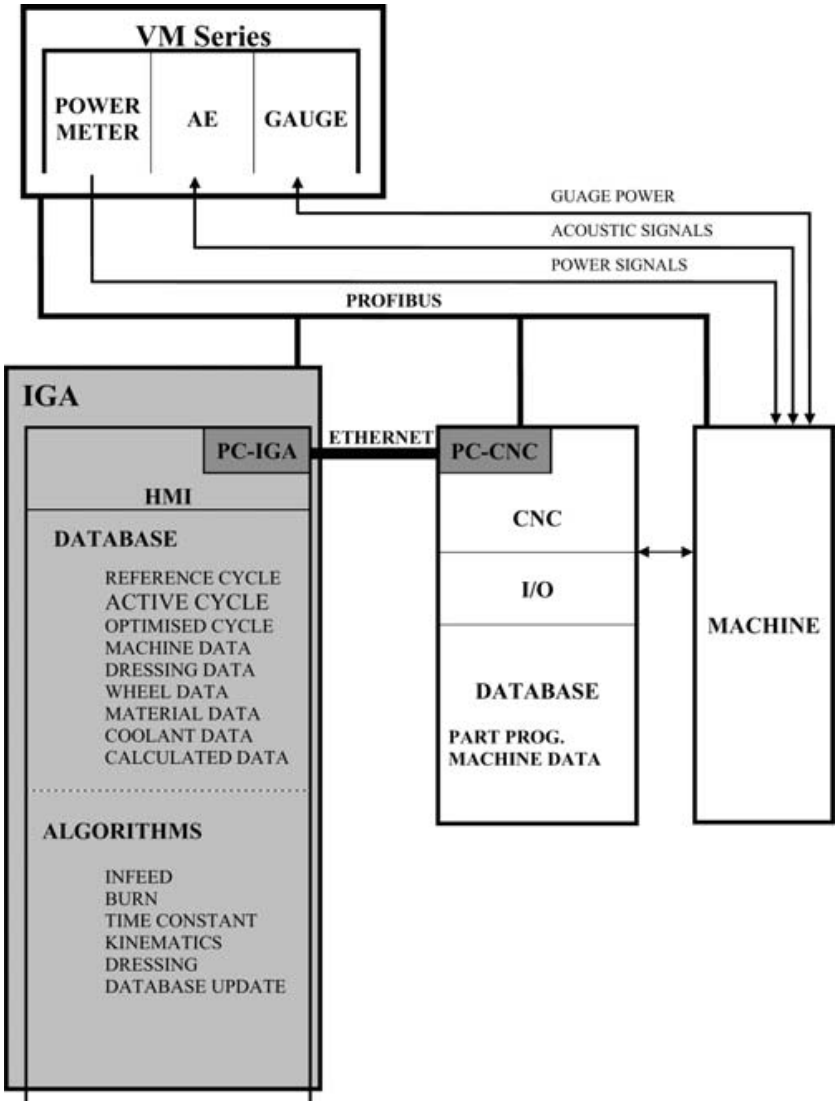

Figure 8 - Morgan et al. Schematic Diagram [4]

3.3 Materials and Ti-6Al-4V in Particular

Material properties, phase transformations and associated microstructures are of importance when machining workpieces. Grinding produces very high temperatures due to the low contact area and high specific energies. The change in temperature has substantial effects on the microstructure of workpieces, with changes in phase changing material properties such as hardness and yield strength. The change in hardness often makes it more difficult to grind the workpiece and thus forces become higher and wear to the grinding wheel occurs at a higher rate. It is thus of vital importance to know what phase the material received is in, as the reliability of the components to be ground can be critical in certain applications.

The system described in this report has initially been designed around the grinding of Ti-6Al-4V, a titanium alloy used notably in aerospace (engines and shuttles), medical, and dental applications [6][7]. It is used in these applications because it has a high resistance to corrosion, low thermal conductivity, high ductility, high strength-to-weight ratio and low density compared to other conventional alloys [6]. It also has a high melting point associated to its low density [6]. Although only this material is described, the same methodology and materials science can be applied to nearly every other workpiece material to acquire similar results of the same significance. To introduce the Ti-6Al-4V material, its properties at various discrete temperatures are shown in Figure 9 below.

Temperature ($°C$)	Density (kg/m^3)	Specific Heat ($J/kg\text{-}K$)	Thermal conductivity ($W/m\text{-}K$)
25	4420	546	7
100	4406	562	7.45
300	4381	584	8.75
900	4294	734	20.2
1100	4267	660	21
1500	4205	732	25.8
1600	4198	750	27
1700	3886	831	83.5
1800	3818	831	83.5

Figure 9 - Ti-6Al-4V Material Varying Properties [46]

Titanium alloys exist in either a hexagonal-close-packed (HCP) crystal structure, a body-centred-cubic (BCC) crystal structure, or most commonly, a combination of the two [6]. The HCP and BCC phases are referred to as the alpha phase and beta phase, respectively. The existence of the two-

phase alloy and its corresponding alpha/beta transformation imply that a large variety of material property variations and microstructures can exist [7]. These variations can generally be achieved through thermo-mechanical processing (namely heat treatment in this scenario) [7]. The phase transformation temperature occurs at 883 degrees Celsius to the beta phase (BCC) and Figure 10 shows the phase transformation diagram of the alloy.

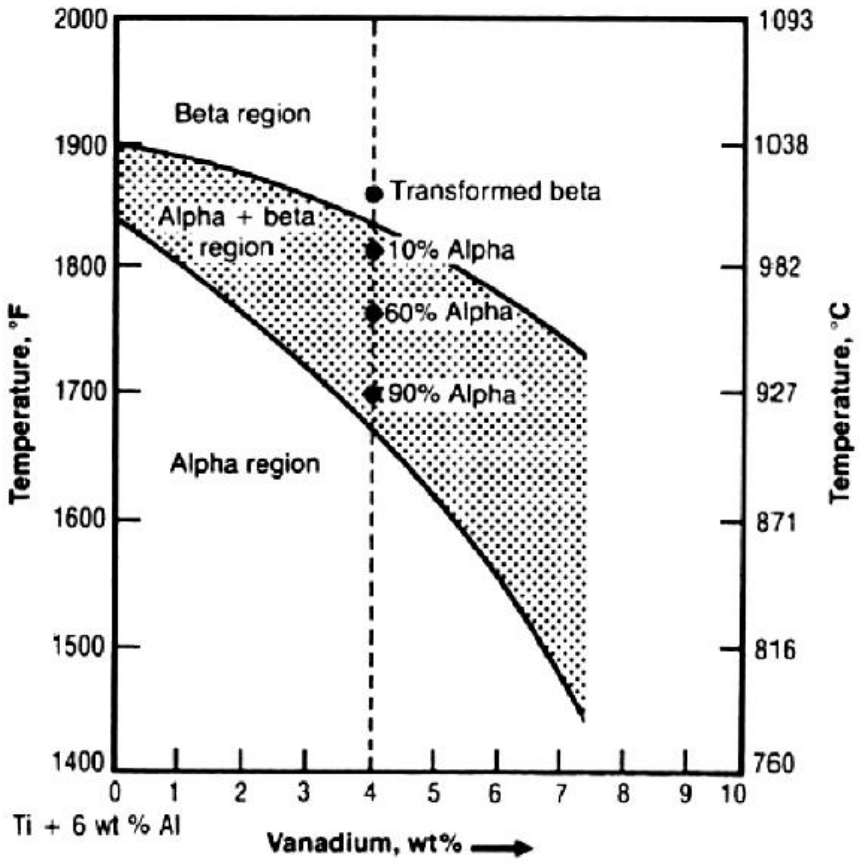

Figure 10 - Phase Transformation Ti-6Al-4V [44]

Different heat treatments result in notably different microstructures once cooled to room temperature. Very slow cooling rates from temperatures around or above the beta transus temperature results in the beta phase transforming into a globular type of alpha, but as the cooling rate is increased, the alpha nucleation rate is enhanced in the beta grain boundaries [7]. Different methods result in vastly different structures and an illustration of the microstructures that occur in the material due to quenching from different temperatures is shown in Figure 11 on the following page.

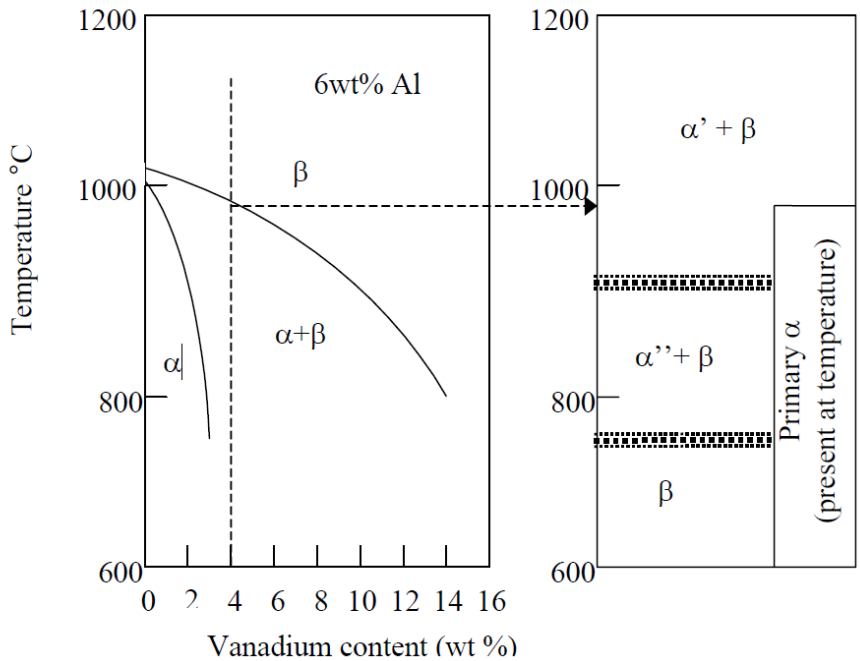

Figure 11 - Phase Transformation Ti-6Al-4V, quenching [7]

Difficulty in machining Ti-6Al-4V is due to many of its inherent material properties [14]. It is very chemically reactive, which leads to higher chip formation and greater tool wear (or premature tool failure) [16]. Its high strength at elevated temperatures and low modulus of elasticity mean it is even harder to machine as well [16]. This means that when developing a system for grinding this material, the changing properties of the material cannot be considered as negligible.

Many other materials exhibit a similar behaviour in terms of phase transformations and property changes with regards to thermo-mechanical processing and if anything, most materials will be easier to machine. Each material will have variations in the manner to which their material properties change with respect to phase, but a model can be made that accounts for the transformative properties and this can be fined-tuned easily as required/if needed.

3.4 Feature Extraction & Selection

Feature extraction is the process of identifying important features and/or parameters in data which are derived from initial sets of data and are intended to be non-redundant and help to facilitate subsequent machine learning [17]. Feature extraction is also involved in reducing the resources required to describe large sets of data and aids in increasing training and learning speed [18].

Liao [19] investigated grinding wheel condition monitoring and sensor-based tool health from acoustic emission sensors, focussing on feature extraction and feature selection. AE signals were collected during grinding operations and these were then processed by DWT (discrete wavelet transformations) or with autoregressive modelling, creating two separate models and finding that the DWT model produces the lowest classification error (7.81%) [19].

Lezanski [20] extracted features from multiple signals, namely those of force, vibration and AE. A feed-forward backpropagation NN was feed with information to select features, which were then used to train a neuro-fuzzy model. Although this varies to what is done in this project as a hybrid duo machine learning technique is used, the concept of extraction and selection before sending the data to a NN for training is similar to what is done in this project.

Kuppuswamy and Airey [21] worked on feature extraction on PCD (polycrystalline diamond) inserts in order to predict product performance and evaluate performance in real-time. Multivariate analysis was used to build a NN and a t-factor was used to assess variability of the data.

Feature extraction is an integral scope of this project and along with this section, the preceding topics contribute to this while the majority of the succeeding topics either fall under this topic or form an output or correlating section to this.

3.5 Modelling

Modelling of a machining process is the first step in building a capable intelligent system. There are many different available techniques to model the grinding system, some of which are more detailed than others. The nature of grinding is non-stationary and complex [13], and thus a reliable model is essential in accurate monitoring. The following reviews will describe mechanisms of modelling used by previous researchers.

3.5.1 Numerical Modelling

The finite element method (FEM) has been a popular technique to model the grinding process and used by researchers such as Hamidi et al [22]. They used FEM to determine residual stresses induced by plane grinding processes, finding that residual stress and temperature increase as wheel speed increases. Another example of FEM used to study residual stress formation is that of Mahdi and Zhang [23], concluding that phase transformation could indicate tensile residual stress at high temperatures.

The use of numerical modelling requires a deep understanding of FEM and would require a significant portion of time and resources that could not be made available in this project.

3.5.2 Experimental/Empirical Modelling

This type of modelling requires a large amount of experimental data to be completed in order to find relationships that can be used to develop empirical factors with reasonable errors, confidence intervals and implied trends.

A typical approach used in this system is that of Werner's force model [14], which has been found to be promising for advanced grinding processes [15] and that the model predicts/estimates forces with acceptable accuracy under specific conditions [15]. The benefit of this model is that once the coefficients and exponents have been found (after many experimental data sets have been collected), the results are accurate for a wide variation of grinding parameters.

Werner's specific force model is given by:

$$F_n = K(C_1)^{\gamma}\left(\frac{Q_w}{v_s}\right)^{2\varepsilon-1}(a_e)^{1-\varepsilon}(d_s)^{1-\varepsilon} \qquad (1a)$$

Where: Fn = Specific normal grinding force

K = Proportionality factor

C1 = Cutting edge density

Gamma = Exponent between 0 and 1 depending on grinding parameters

Qw = Specific material removal rate

vs = grinding wheel speed

Eta = exponent between 0.5 and 1 depending on workpiece material

ae = depth of cut

ds = diameter of grinding wheel

Equation (1a) can be used to estimate forces in grinding for more machinable materials (eta = 1) and less machinable materials (eta = 0.5) [25].

Mishra and Salonitis [25] produced a modified version of Werner's force model which replaces the above proportionality factor K, grinding property exponent gamma, and cutting-edge density C1 by a single factor, K1. This means only two empirical factors were needed, K1 and eta.

The modified Werner's equation is given by:

$$\left(F_n\right) := K_1 \cdot \left(\frac{Q_w}{v_s}\right)^{2\varepsilon-1} \cdot a_e^{\,1-\varepsilon} \cdot d_s^{\,1-\varepsilon} \tag{1b}$$

Through experimental data collection, this model was proven to be within accuracy limits [25].

This approach would have been taken if it were possible to complete experiments during the year as access to equipment would have been available and experimental data would have had benefits for the machine intelligence section as well as the wheel wear analysis and temperature analysis. Obviously due to the limitations brought upon us by the Covid-19 pandemic, a theoretical approach needed to be taken that would also be reasonable enough to complete in the time frame given, with the resources available, and the knowledge known by a final year undergraduate mechanical engineer.

3.5.3 Micro-Analytical Modelling

There are two types of analytical modelling available when describing the grinding process from a mathematical perspective. The analytical process has been the preferred process as it provides insight from both a physical and theoretical sense [13]. To date, a comprehensive micro-model (looking at the grain mechanisms) has not been created, but models that incorporate some mathematical or empirical aspects have shown positive results.

Jamshidi and Budak [24] created a force model based on individual grit interaction, creating a topography of the grinding wheel related to its grits and a topography of the workpiece relating to surface roughness. Their algorithm can be seen in Figure 12 below.

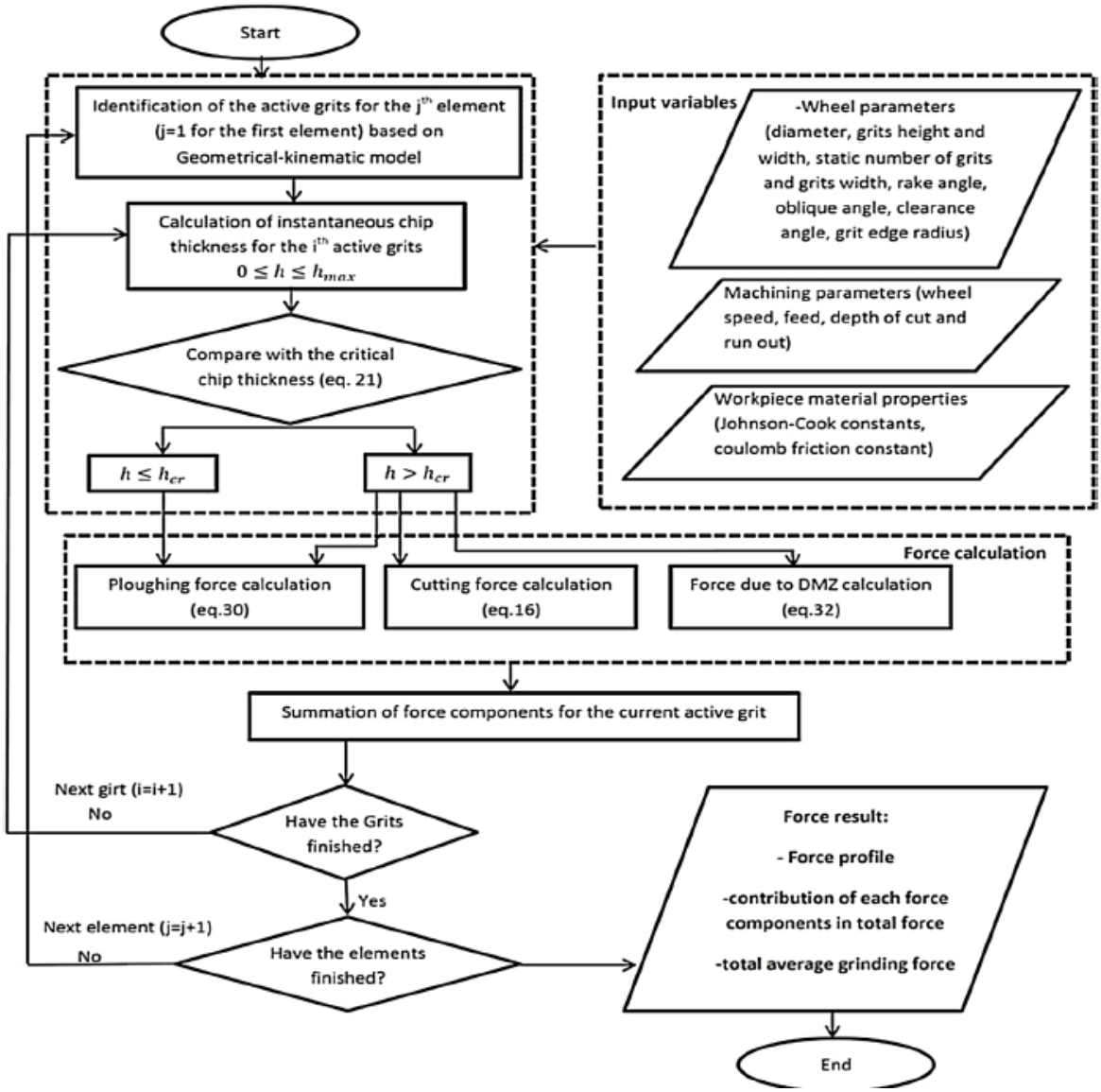

Figure 12 - Flowchart for Grit Modelling (Micro) [24]

The modelling and interaction of grains is a very in-depth process and is a project of its own. Thus, it has not been selected as the method of modelling.

3.5.4 Macro-Analytical Modelling

Once again, this method is based on the fundamentals of physics, but here the process is looked at as a whole, incorporating overall grinding parameters. These incorporate many of the classical manufacturing equations and knowledge described in textbooks, taught through the path of undergraduate mechanical engineering.

Notable analytical models are that of Tonshoff [25] who modelled grinding induced residual stresses and Chen et al [26], who explained the transitional temperature between tensile and compressive residual stresses [13].

Due to its theoretical nature, reasonable accuracy and ability to finish in a reasonable timeframe, the macro-analytical approach was used to create a model for this system. The development of it will be described in further sections of the literature review as well as in the body of the report following the review.

3.6 Residual Stress, Temperature and Coefficient B

Abrasive grinding, which is the final processing stage for many components, induces residual stresses at the surface and sub-surface of the material [27]. This will affect the service life of the component as well as its reliability [27]. Grinding conditions which are harsher and more abusive to the component generally increase the thermally induced damage to the workpiece [26]. Note, as stated earlier, that an increase in temperature while grinding has been correlated to an increase in residual stress accumulation in the workpiece. It is also true that the onset of tensile surface residual stresses is caused by exceeding a critical transition temperature [27].

Residual stresses exist due to complex thermo-mechanical interactions between the grinding wheel and the component [23]. The thermal field has a considerable effect on the presence and magnitude of apparent residual stress [29]. When materials are exposed to high temperatures, the surface materials expand, and yield strength is lowered, causing an unrecoverable plastic deformation in the object's surface layer [29]. Then when the material cools (often quite rapidly), the surface layer shrinks, and tensile residual stresses become apparent and, as stated above, can cause detrimental effects to its service life.

The greatest source of energy used in the process is that of heat dissipated into the workpiece (which can cause severe damage to surface integrity) [4]. Although it is quite a difficult parameter to measure, residual stress is an important parameter to consider when assessing the surface integrity of ground components. Temperature increase is thus important to measure due to its direct effects on residual stresses.

The development of a model that focusses on residual stress and temperature as important parameters (workpiece stress not being exceeded, and workpiece burn not occurring) will be a main output of the feature extraction stage.

As seen in Figure 5 on page 20 that Bhusan created, the main signals that can be measured in real time during grinding are: force, acoustic emission (AE), power, and vibration [28]. It would therefore be beneficial that one of these signals which can be easily measured in real time could describe residual stress in some way.

Kruzsynski and Wojcik [30] created an empirical model that modelled residual stress in grinding. They related residual stress to a formula they named the 'Coefficient B' and found that there was a linear relationship between the two, implying that if one could calculate the coefficient, one could also know the residual stress state of the component. Coefficient B is a simple equation that takes

the form of power density multiplied by contact time, or total power divided by width of cut and work speed.

The coefficient B is given by:

$$B = P't_c = \frac{P}{b_c l_e}\frac{l_e}{v_w} = \frac{P}{b_d v_w} \tag{2}$$

P' = power density, tc = contact time, P = total power, bd = width of cut, vw = workspeed

Figure 13 and Figure 14 below show the results acquired by Kruszynski and Wojcik [30], which show the relationship between residual stress and coefficient B for alloy steel (Figure 13) and for bearing steel (Figure 14).

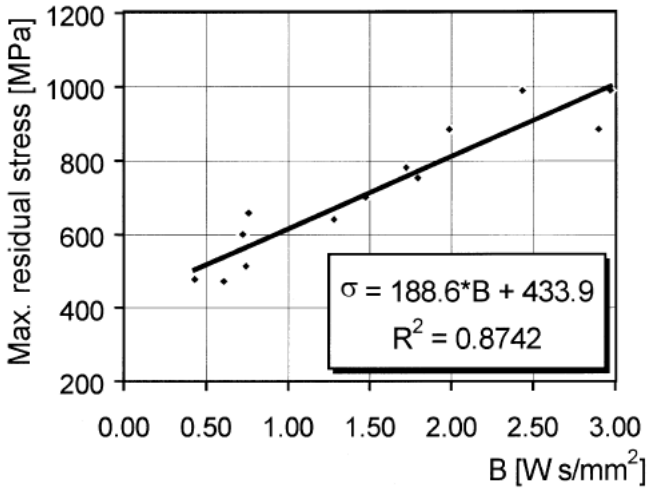

Figure 13 - Alloy Steel - Residual Stress vs Coeff. B [30]

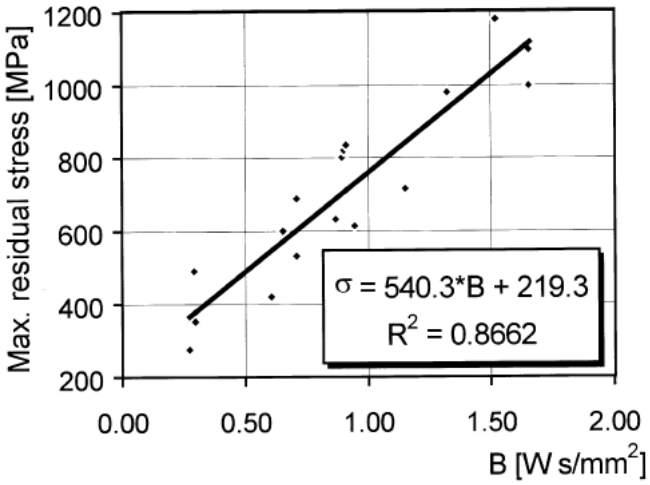

Figure 14 - Bearing Steel - Residual Stress vs. Coeff. B [30]

3.7 Signal Processing & Noise Elimination

Raw signals acquired during the grinding process are often weak and contain noise generated by electronic devices, bond extrusion and many other possible sources [31]. These are obviously unwanted and so the signals need to be processed and noise needs to be eliminated.

Low-pass (LP) and high-pass (HP) filters remove the low-frequency and high-frequency noise components to avoid aliasing [31]. Mirifar et al [31], processed AE signals and applied a LP and HP filter to signals using MATLAB's toolbox. The change in signal can be seen in Figure 15 below.

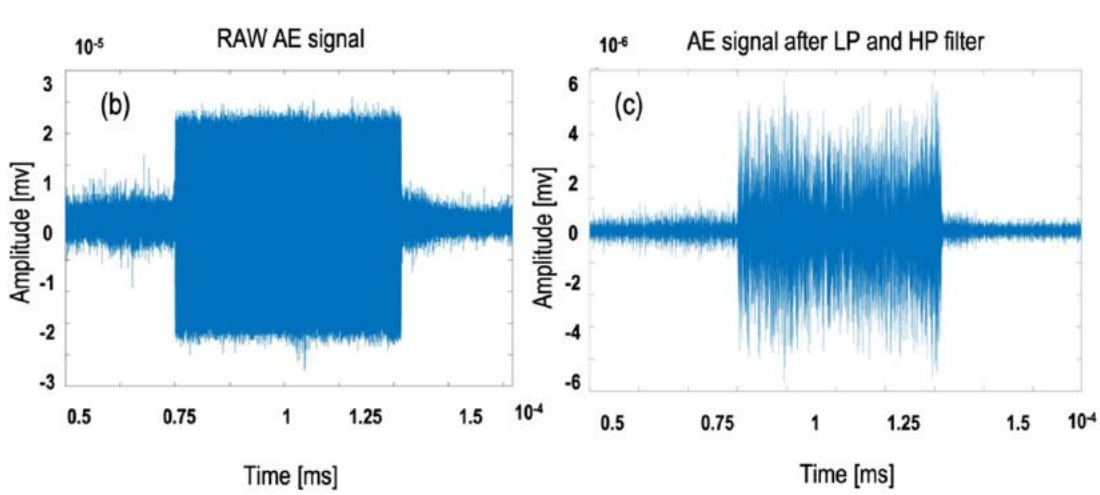

Figure 15 - LP and HP Filters [31]

After this processing, they used discrete wavelet transformations to decompose the signals and eliminate noise. The DWT provides a multi-resolution representation of a signal using discretely sampled wavelets. This is very useful in image processing and in this case, to eliminate noise. Figure 16 below shows the AE signals that have had their noise eliminated.

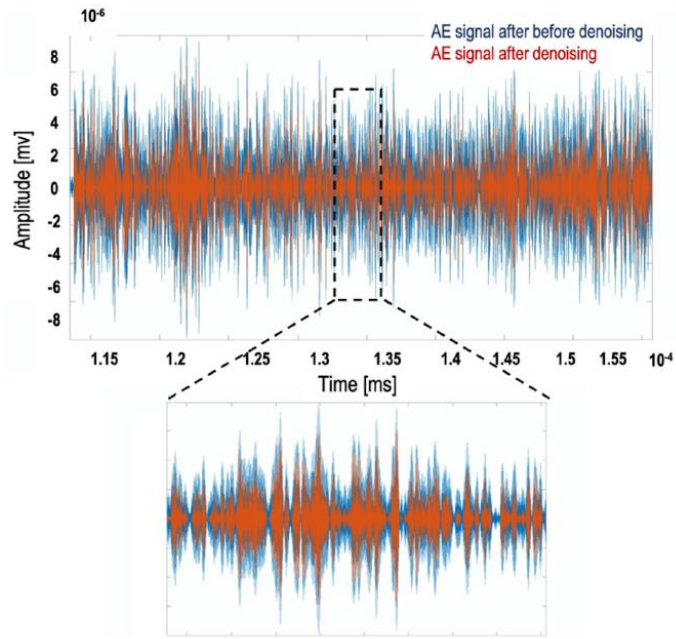

Figure 16 - AE Signal Denoising [31]

A flowchart of their logic is shown below:

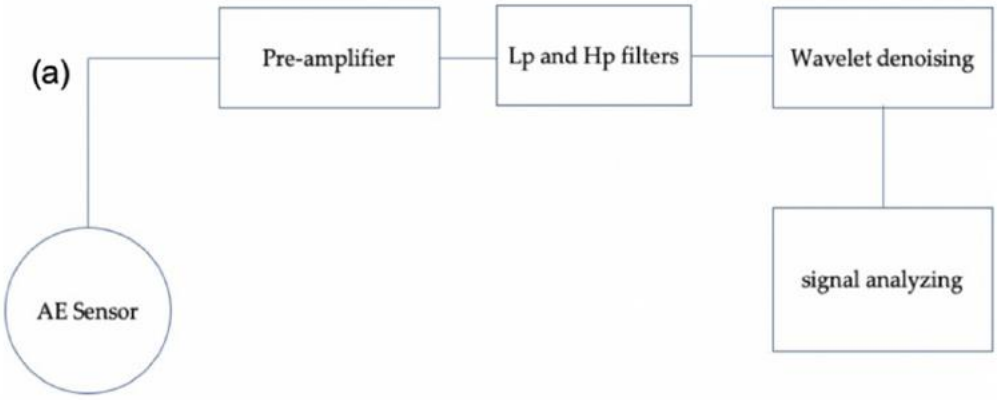

Figure 17 - Denoising Logic [31]

Another viable method of denoising signals is that of a moving average filter, which smoothens peaks and valleys of signals by taking time-sampled averages of signals over a specified range and producing a new signal after every time step.

3.8 Wheel Wear/Life and Kurtosis Coefficient

Using the correct wheel when grinding is of utmost importance and can make the process more effective and economical [32]. Grinding ratio describes the ratio of volume of work removed to volume of wheel removed, with higher values implying higher wheel life. Nearly all grinding parameters affect wheel life, but wheel diameter has possibly the greatest effect on wear [32]. Kalaszi describes the effect of wheel wear in his paper [32] as well as proposing an empirical model that predicts wheel life (using a simple ratio based on grain size distribution and conditions of wheel dressing, which are determined by wear experiments).

Abrasive wear is the most apt description of grinding wear. It is a serious type of wear and occurs when two surfaces interact in direct physical contact [34]. Figure 18 below shows an abrasive wear model.

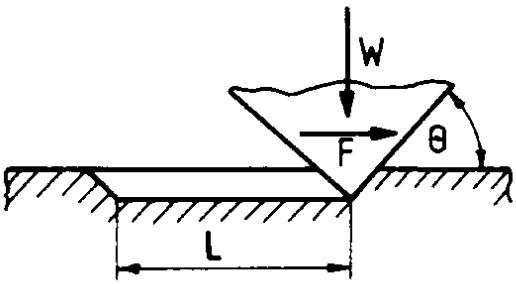

Figure 18 - Abrasive Wear, Tribology [47]

The following simplified formula shows the volume of abrasive removed for a length of workpiece removed:

$$V_{abr} = \frac{2}{\pi}\frac{\tan(\theta)}{H}WL \tag{3}$$

With Vabr = volume of abrasives removed

Theta = angle of inclination of tool

H = Hardness of workpiece

W = Normal grinding force

L = Length of workpiece

The kurtosis coefficient measures the skewness of a curve and can aid in the analysis of tool life monitoring. This was done by Nuawi et al. [33], who used integrated kurtosis-based algorithms to predict tool life by finding a trend of the coefficient for flank wear of a cutting tool. The conclusion was an effective tool life prediction means. The general forms of kurtosis (describing forms varying away from a kurtosis coefficient of 0) can be seen in Figure 19 below.

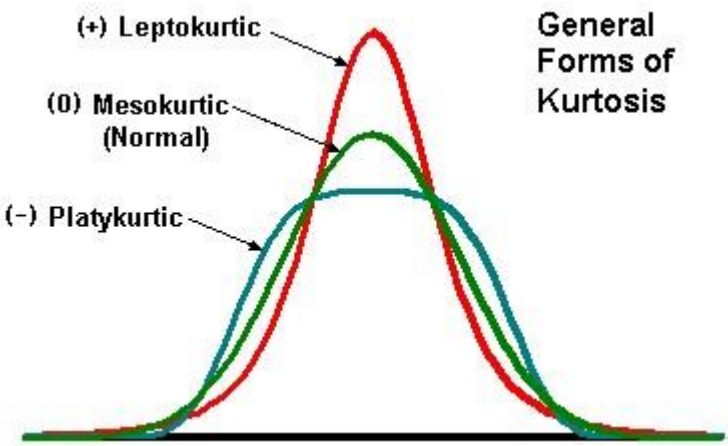

Figure 19 - General Forms of Kurtosis [45]

3.9 Database

Storage and consequent access of data is fundamental to the development of a successful intelligent system. The most practical manner to do this is the use of a database. In the development of Morgan et al.'s IGA system [4], a separate paper was written describing just how the database was developed [34].

The system was developed in Microsoft Access with the use of Visual Basic support code, and includes data for cycle, material, wheel, dress, coolant, machine and part records. It links to a CBR (case-based reasoning) and RBR (rule-based reasoning) to aid the IGA in selecting grinding parameters. A front-end graphical user interface (GUI) for the database was developed in order to easily add, edit and remove information.

Choi and Shin's GIGAS [8] contains two separate databases for machine data and process models, respectively. Although each system will have a differently structured database, all will need this method of storage for the system to implement learning and 'intelligence'.

3.10 Human-Machine Interface

The human-machine interface, also known as the operator's graphical user interface, provides a means of viewing information and a connection between the human using the system and the grinding process, as well as background algorithms and models.

The development and structure of GUIs has remained the same since the implementation of intelligent machining systems. It contains a starting page that navigates you to the input page where a user chooses certain grinding parameters relevant to the system e.g. workpiece material size, surface requirements. After the inputs a background (non-visible) connection to algorithms and the database is made, and then a front-end output is shown to the user which will either give relevant recommendations on what the operator should do to optimize the process (or in more modern systems, it will be implemented automatically, as there is a direct connection between the intelligent system and the CNC). In a live system (of which nearly none have been developed), the outputs will be taken to the running of the system and the algorithms will run while the grinding process is in place, ensuring that the process runs as intended.

A main menu GUI from W.B. Rowe et al.'s [35] generic intelligent grinding system (the first recognised intelligent grinding system) can be seen in Figure 20 below.

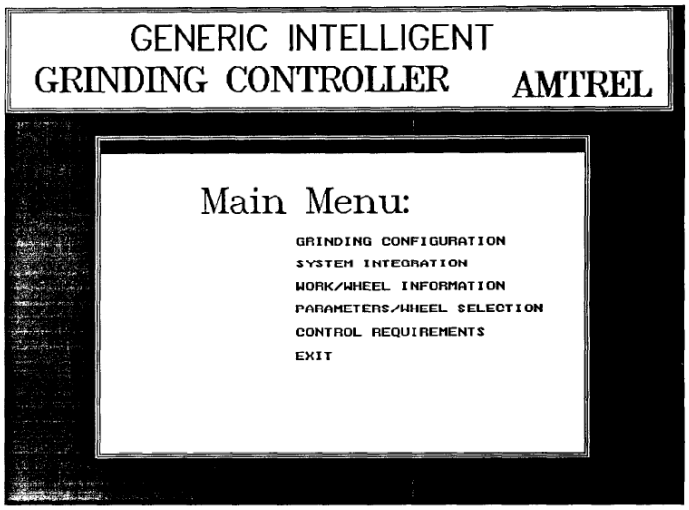

Figure 20 - Rowe et al. GUI [35]

The output/recommendation's page from Morgan et al.'s [4] intelligent grinding system can be seen below in Figure 21.

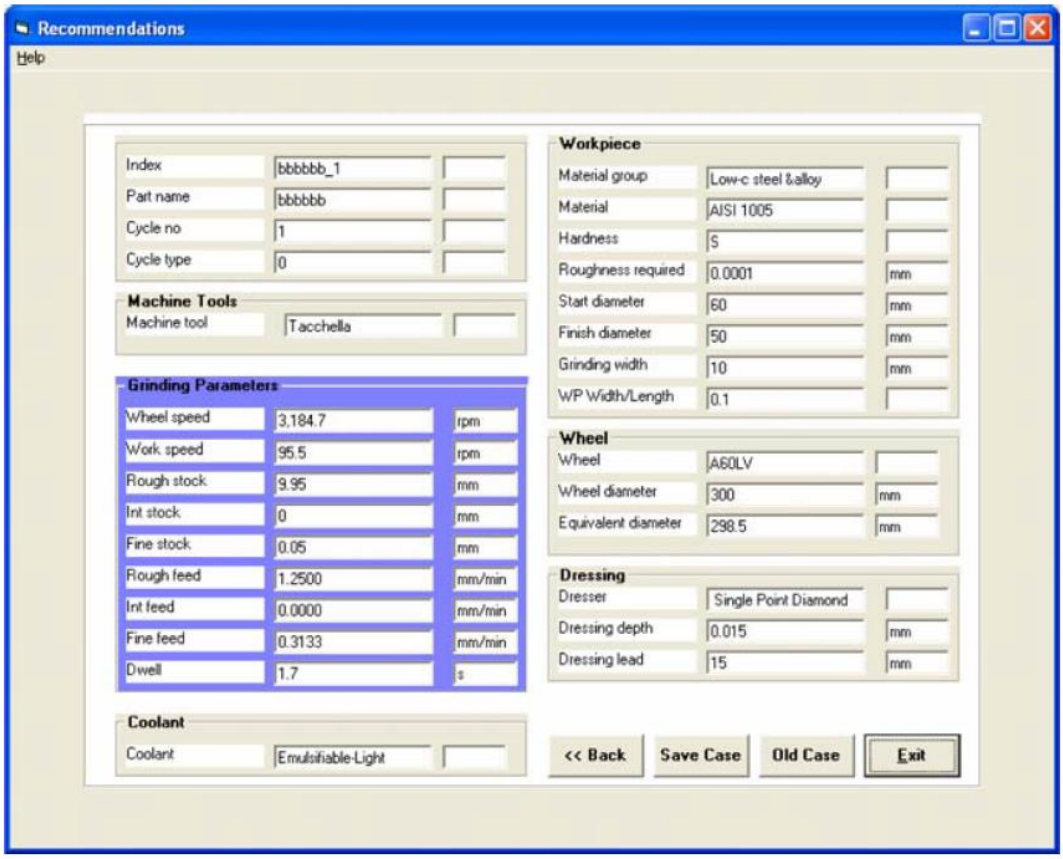

Figure 21 - Morgan et al. GUI [4]

The connection of interfaces for Choi and Shin's GIGAS [8], can be seen in Figure 22 below.

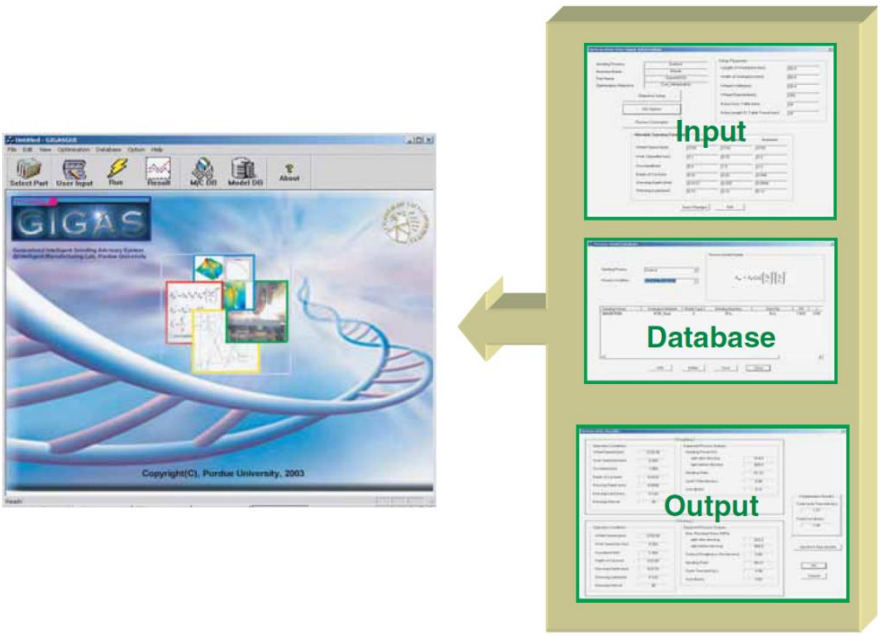

Figure 22 - Choi and Shin Interfaces [8]

3.11 Feature Correlation & Machine Intelligence

Although this is not part of my scope and is rather part of the concurrently worked project of Kalvin Govender, an acknowledgement and short review of it is required in order to effectively understand how the system works (as the implementation of machine intelligence is what makes a system intelligent instead of a simple input and output).

Feature correlation follows directly from feature extraction and makes use of statistical methods to determine the most important parameters to select based on what the output should be. Following this, the artificial intelligence model can be built to develop a system that can effectively predict outputs. However, preceding the development of the model, a method of prediction must be chosen.

Many different types of artificial intelligence (prediction systems) are available to be used and many have been used in the development of machine intelligence systems, with neural networks being a popular choice.

Lezanski [10] created an artificial neural network based model to predict grinding wheel wear (and remaining wheel life), while Zhao et al. [36] used a fuzzy logic method as a prediction method for their control system. Morgan et al. implemented a mixture of neural networks, case-based reasoning

and rule-based reasoning in the development of their intelligent grinding system [4]. Kuppuswamy and Airey analysed signal relationships through multivariate analysis in order to create a multi-layer perception neural network for prediction of flank wear on poly-crystalline diamond inserts [21]. See Figure 23 below for the neural network model developed by Kuppuswamy and Airey [21].

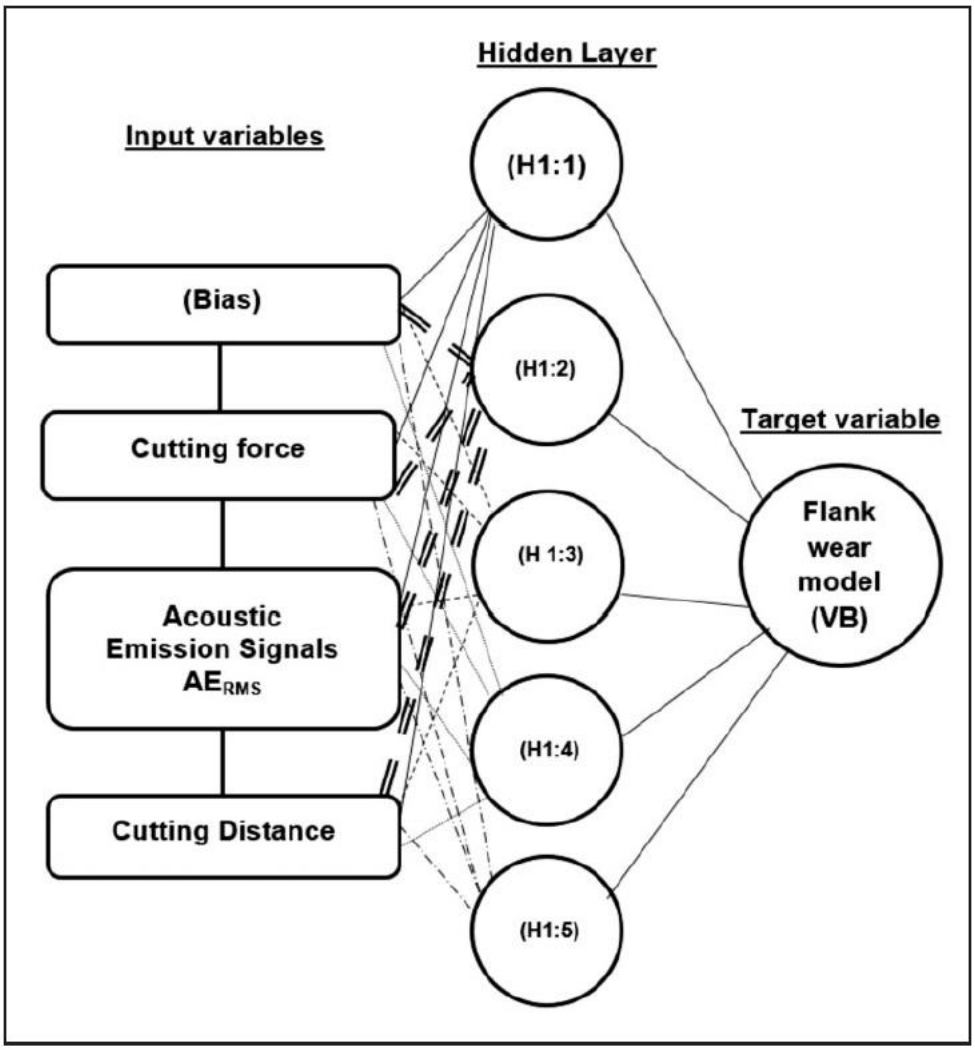

Figure 23 - Kuppuswamy & Airey - Neural Network [21]

3.12 Summary of Literature Review

The literature survey presented above details the previous work done on intelligent grinding systems as well as the necessary theory and science behind materials, artificial intelligence, modelling of the grinding process, interfaces, data storage, usage of statistical knowledge, scientific methods, physics based knowledge and signal processing, required to develop an intelligent grinding system. The aspects of the literature survey that are of most importance to the development of this project will be stipulated in the summary below.

All the fundamentals of grinding and background on the physics based process, where grinding parameters are concerned, have been used and extensively implemented in the project, this includes aspects such as wheel and work speeds, depth of cut, diameter of the wheel, workpiece size, specific grinding energy and coefficients of friction as well as all their interconnected relationships.

The material properties of workpieces (and wheels to some extent) are considered, particularly noting phase transformations, hardness and yield strength for the development of a force model, while many other material properties e.g. density and coefficient of thermal expansion, are used for the development of other portions of the model (temperature and residual stress, or power, are just some examples).

The use of semi-empirical/statistical and analytical models are enforced in this paper and the development is highlighted by the accurate determination of force, coefficient B, temperature and residual stress, with other outputs an added bonus that comes as a part of calculating the aforementioned.

The discrete wavelet transformation and moving average filter are the main takeaways from the review and are implemented in the project.

Wheel wear using the empirically developed model stated as well as the statistically developed kurtosis coefficient are important features of the literature review to take note of.

Database and graphical user interface development and their structures as well as linkages to each other and the operator, are of most importance when concerning the literature review.

4 Methodology

4.1 Feature Extraction

The following section describes the development of the feature extraction engine, which includes sections for the following: modelling, mock data creation, experimental data collection, statistical analysis, noise elimination and description of response surface methodology outputs.

Please note that the following software packages are required to run the developed models, database and human machine interfaces:

MATLAB R2020a and the following sub-packages:

App Designer

Wavelet

Database Explorer

(note that a connection between the database and MATLAB must be made, and code must be changed as per operated computer)

(In the Analysis Page, the source code for loading data will need to be changed too)

Microsoft Access 2006-2017 version

DeweSoft X3

(Optional: Microsoft Excel, to see the initial model and experimental results)

4.1.1 Analytical/Semi-Empirical Modelling

This section is of most importance to the feature extraction engine and contains the majority of work done for this section of the project. Creation of the initial model (as created in the interim report) will be discussed first, before detailing the development and completion of the final model, which is used in the intelligent grinding system.

To create a model, process parameters are required, as well as characteristics of the interacting objects. For developing a grinding model, the initial known process parameters (set by the operator) are: work speed (feed-rate), wheel speed, and depth of cut. This is obviously in combination with the selected grinding wheel and workpiece. From the selected grinding wheel we can acquire the grinding wheel diameter and width, as well as the grinding wheel characteristics: mesh number, grade, structure, abrasive layer thickness, abrasive concentration and wheel hardness. From the

workpiece material we can acquire all known material properties. Important material properties are: Brinell hardness (BHN), density, yield strength, elastic modulus, specific heat capacity, coefficient of thermal expansion, thermal conductivity, phase transitional states (microstructural information) and phase transitional temperatures. Many of these properties are measured at room temperature and vary with an increase in temperature when grinding. This information has been experimentally obtained for many materials and thus the changing properties can easily be accounted for without the need to complete many experiments (which would not have been possible in these circumstances).

The initial model was created on Microsoft Excel and followed the techniques of abrasive machining, focussing mainly on physics based analytical formulas, with certain empirical formulas which form the heart of the output information, sought for. Figure 24 below shows the logic followed to acquire the output force, thermal, and residual stress characteristics of the process, as well as the characteristics that are not necessarily desired outputs but can be derived along the path to output. The derived values (such as surface roughness, grinding force ratio, contact time and grain force) are useful in optimization and can also be used for operator specification. Two examples of this would be: optimizing for least amount of wheel wear and optimizing the process for a specified surface roughness according to the user's input.

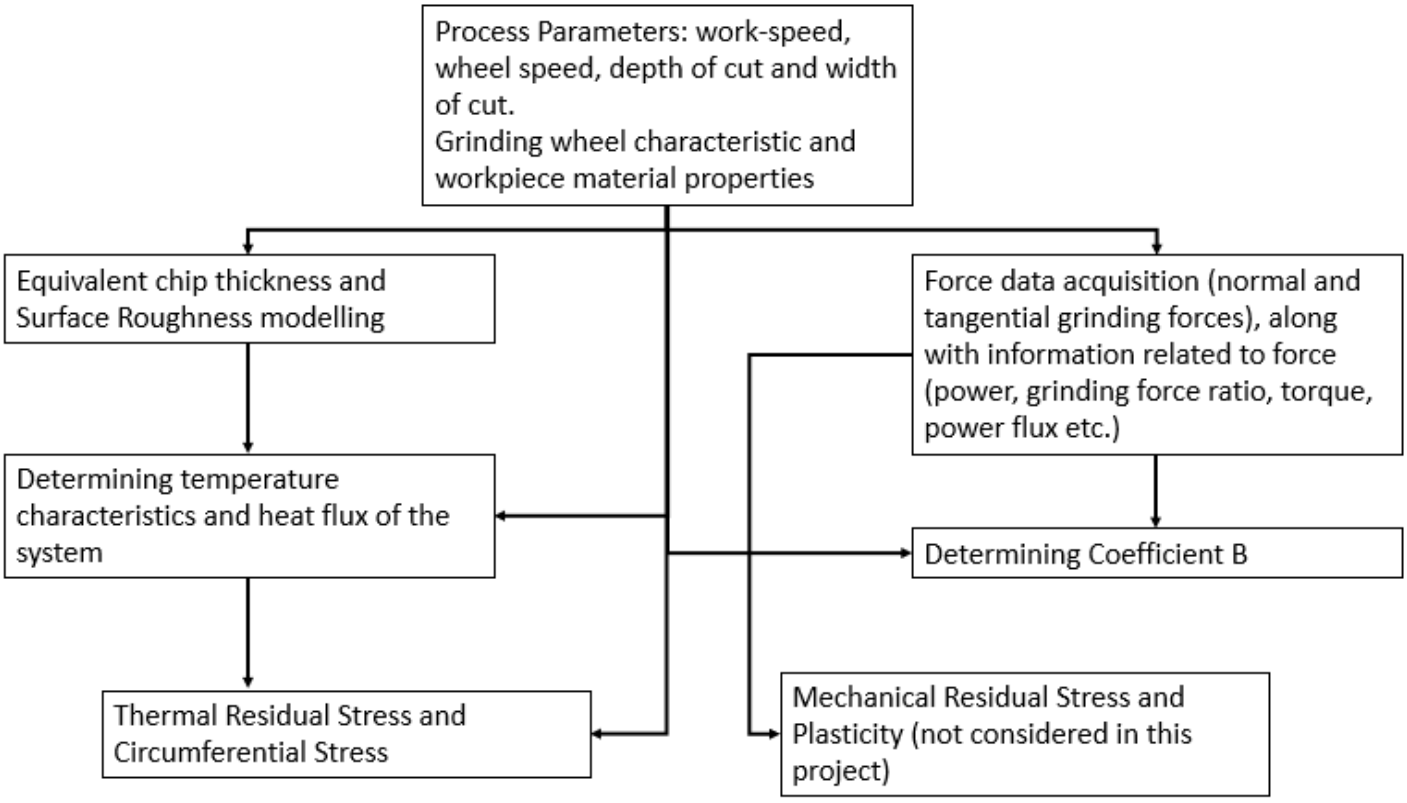

Figure 24 - Initial Model Flowchart

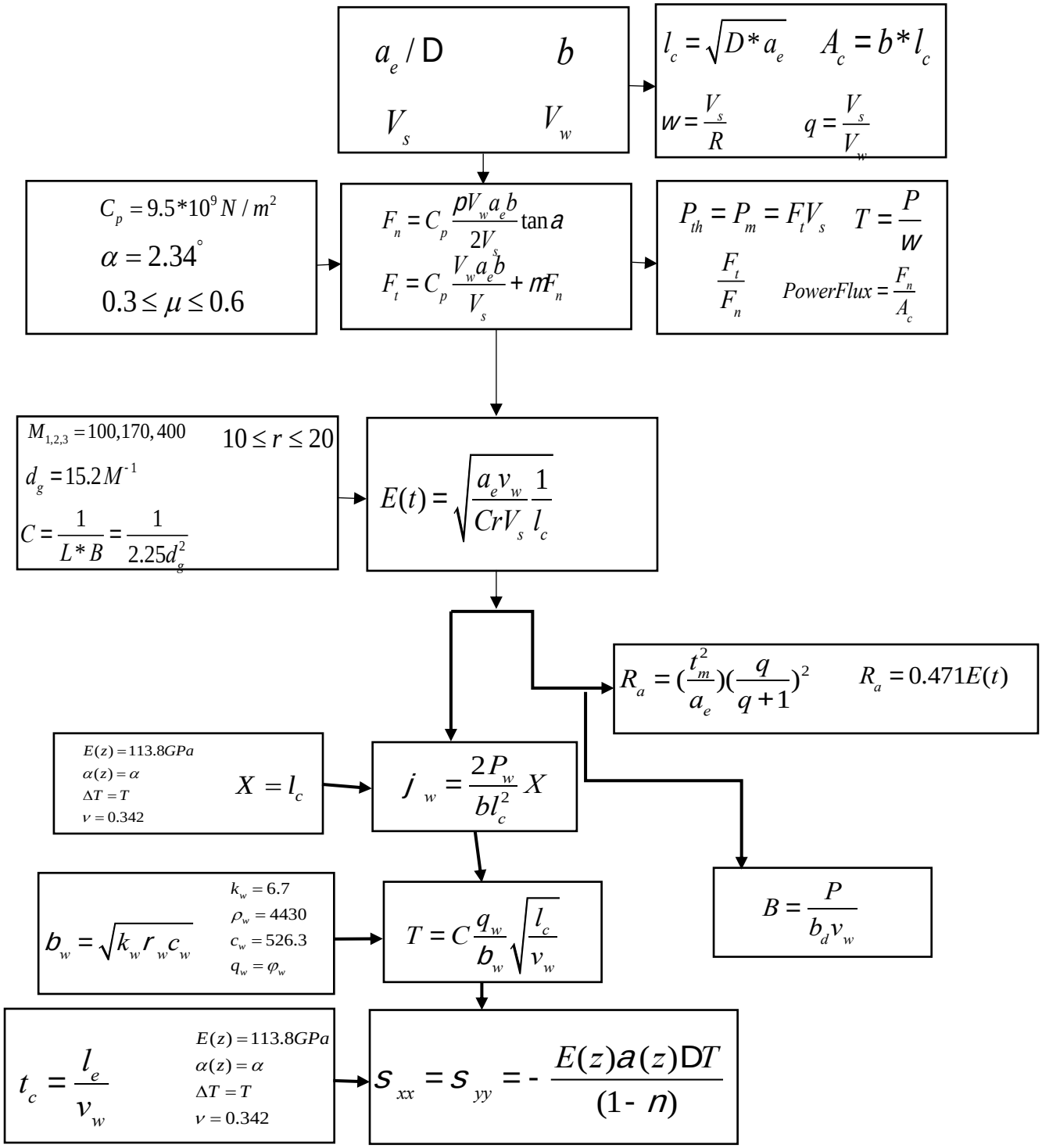

Figure 25 - Initial Model Flowchart (Formulae and Parameters)

Figure 25 on the previous page displays the same information as Figure 24, however it shows all equations and assumptions made rather than the sentence-typed logic. Note that the appended values for material properties are relevant for Ti-6Al-4V but any material properties could be put there in place of these.

The initial model begins by taking work speed, wheel speed, depth of cut, width of cut (wheel width), wheel characteristics and material properties as input. This then allows for the calculation of length of cut, torque, contact area and grinding speed ratio as nominal outputs.

Based on empirical or known previous data, the normal force and tangential force can be calculated by the following formulas:

$$F_n = C_p \frac{\pi V_w a_e b}{2V_s} \tan(\alpha) \tag{4a}$$

$$F_t = C_p \frac{\pi V_w a_e b}{V_s} + \mu F_n \tag{4b}$$

Where:

C_p = Specific Grinding Energy

V_w = Work Speed

a_e = Depth of Cut

b = Width of Cut (Width of Wheel)

V_s = Wheel Speed

α = Included angle of grain (wheel property)

μ = Coefficient of Friction (dependent of the interaction between wheel and workpiece)

This empirical formula is described in the PhD thesis by Inada [37], where many high-speed grinding experiments were completed to confirm the accuracy of the model. These two formulas are pertinent to the development of the model as most succeeding formulae require tangential force (and thereby normal force) as an input in some regard.

Specific grinding energy (or specific cutting energy) is a measure of the ability of a grinding wheel to remove material and depends on the grindability of the workpiece and sharpness of the wheel [38] and is generally found through experimentation and validation. However, research has been completed on a large quantity and variety of wheel and workpiece interactions and thus the specific grinding energy is known for most interactions that will take place, thus making the empirical formula suitable for the intelligent grinding system developed. Note that if experimental work could have taken place during the completion of this project, the values of specific energy would have been acquired for many interactions and verification would have been a part of this project.

The specific grinding energy is given by:

$$C_p = \frac{P}{Q} \tag{5}$$

Where P = total grinding power and Q = material removal rate.

Note that Q can be calculated using the parameters stated above and does not need to be found experimentally, where P would generally be the value found experimentally.

$$Q = a_e b v_w \tag{6}$$

From the tangential force value acquired, coefficient B (as described in the literature review), can be found. Coefficient B has been validated to have a linear relationship to residual stress for a large variety of grinding conditions [30]. This output allows for an initial prediction model to be built based on the required output of no over-stressing occurring. However, without a residual stress estimation, no comparison can be made for the Coefficient B, and thus limits cannot be applied to describe if a workpiece has been overstressed or not.

From literature and review it was noted that the residual stress induced in a workpiece surface by grinding is directly proportional to the temperature rise in the workpiece, given by:

$$\sigma \propto \Delta T$$

After more research and review, the general form for residual stress after a temperature change, was found [39]. Becker et al derived the following formula as an approximate method for calculation of residual stress in functionally graded materials (materials whose properties change with respect to their dimensions):

$$\sigma_{xx} = \sigma_{yy} = -\frac{E(z)\alpha(z)\Delta T}{(1-\nu)} \qquad (7)$$
$$\sigma_{zz} = 0$$

Where:

$\sigma =$ Residual Stress (subscripts indicate plane)

$E(z) =$ Elastic Modulus (Young's Modulus)

$\alpha(z) =$ Coefficient of Thermal Expansion

$\Delta T =$ Change in Temperature

$\nu =$ Poisson's Ratio (material property)

The change in dimension 'z' would remain constant in non-functionally graded materials and is still an acceptable means of residual stress estimation as functionally graded materials are a special case of general materials whose properties change with varying dimensions. The thermal field is by far the most considerable factor influencing the surface of workpieces when grinding (as this is the accumulation of both thermal stresses and phase transformation) [29]. Thus, other means of residual stress accumulation were not considered.

Due to the fact that one needs to determine the temperature rise to estimate residual stress, a model of temperature increase in grinding was required. The most generally accepted method for determining temperature increases [26] in grinding is given by:

$$T = C\frac{q_w}{\beta_w}\sqrt{\frac{l_c}{v_w}} \qquad (8)$$

Where:

$C =$ Constant depending on heat flux pattern distribution

$q_w =$ Heat flux entering the workpiece

$\beta_w =$ Coefficient of heat diffusion

$l_c =$ Contact length

$v_w =$ Work speed

Heat flux pattern distribution constant can generally be taken as 1 when considering a triangular heat source (which is the case for this assumption) [26].

The coefficient of heat diffusion is a root of a product of three workpiece materials and has very little variation over high variations of temperature, making it a suitable constant to use when calculating temperature rises [26]. This coefficient is given by:

$$\beta_w = \sqrt{k_w \rho_w c_w} \tag{9}$$

Where:

$k_w =$ Thermal conductivity of the material

$\rho_w =$ Density of the material

$c_w =$ Specific heat capacity of the material

Contact length in surface grinding is simply the root of the product of the wheel diameter multiplied by the depth of cut, given by:

$$l_c = \sqrt{D.a_e} \tag{10}$$

While heat flux into the workpiece is simply given by:

$$q_w = \frac{2P_w}{bl_c} \tag{11}$$

$P_w =$ Power used to create heat in the workpiece (approximately 65% of the net power)

$b =$ Width of cut

$l_c =$ Length of contact (as above)

Lastly, the grinding power can be calculated simply as:

$$P = F_t.v_s \tag{12}$$

Where:

$F_t =$ Tangential force (as calculated previously)

$v_s =$ Wheel speed

The last two important parameters that can be found are that of chip thickness and surface roughness. The chip thickness is required to calculate surface roughness and is used to relate conformity of grinding contact to a particular abrasive [40].

To acquire a value for chip thickness, properties of the wheel must be known, namely the mesh number of the wheel (M).

Knowing this number allows one to calculate the equivalent spherical diameter of abrasive (diamond) particle by:

$$d_g = \frac{15.2}{M} \tag{13}$$

With M increasing in number as coarseness of the wheel decreases (with numbers around 10 being very coarse and numbers around 600 being very fine).

The number of cutting points per unit area of periphery of the cutting wheel can then be found (empirical formula) by:

$$C = \frac{1}{2.25 d_g^{\,2}} \tag{14}$$

The last parameter needed to calculate chip thickness is the ratio of chip width to undeformed thickness (known as r). This is estimated to be between 10 and 12 for the surface grinding process.

Undeformed chip thickness is then given by:

$$E(t) = \sqrt{\frac{a_e.v_w}{C.r.v_s}\frac{1}{l_c}} \tag{15}$$

Where all the variables contained in this formula are described above.

This then allows a simple empirical calculation of surface roughness as:

$$R_a = 0.471E(t) \tag{16}$$

Other outputs from the model were found but were not used later in the development of the intelligent system and were thus unnecessary to describe here, but they are contained in the empirical model flowchart in Figure 25 on page 44.

This concluded the initial development of the model and provided a basis on which to develop the feature correlation engine for the other team member while a more descriptive and accurate model was prepared (by modifying the initial model).

4.1.2 Final Empirical Model Used in the System

The initial model created contains all the outputs needed to create the intelligent system. As with any model, the scope to which one can attempt to simulate is endless but achieving a model capable of prediction by limiting the variables which affect the model to those of most importance, and of most affect, is a fine line. The initial model is reasonable but misses a few key aspects and outputs necessary to satisfactorily represent the system.

The missing aspects, which are accounted for in the final model and are described below, consist of: accounting for phase transformation of the material when higher temperatures are reached, accounting for size of the workpiece to incorporate a time based model (multiple passes of the grinding wheel), accounting for temperature dependent material properties, and incorporating a model describing the wear of the wheel (which should be incorporated into a dress cycle optimization or wheel replacement prediction model). The only features excluded from the model are the incorporation of coolant effects on the system and a dress cycle model, as these would have taken too much time to create and would have not allowed for timeous completion of the project. However, these could easily be incorporated into the model at a later stage and accommodations have been made in the database and GUI so that when developed, the coolant and dress cycle could easily be integrated into the system.

Figure 26 on the following page shows a flowchart of the final model created and provides the basis on how the database was developed. The model was created on MATLAB, making use of various toolboxes and functions. The associated code for the model can be found throughout the system app (inside the code view of the app designer) as well as in the Appendix, where it has been placed for reference.

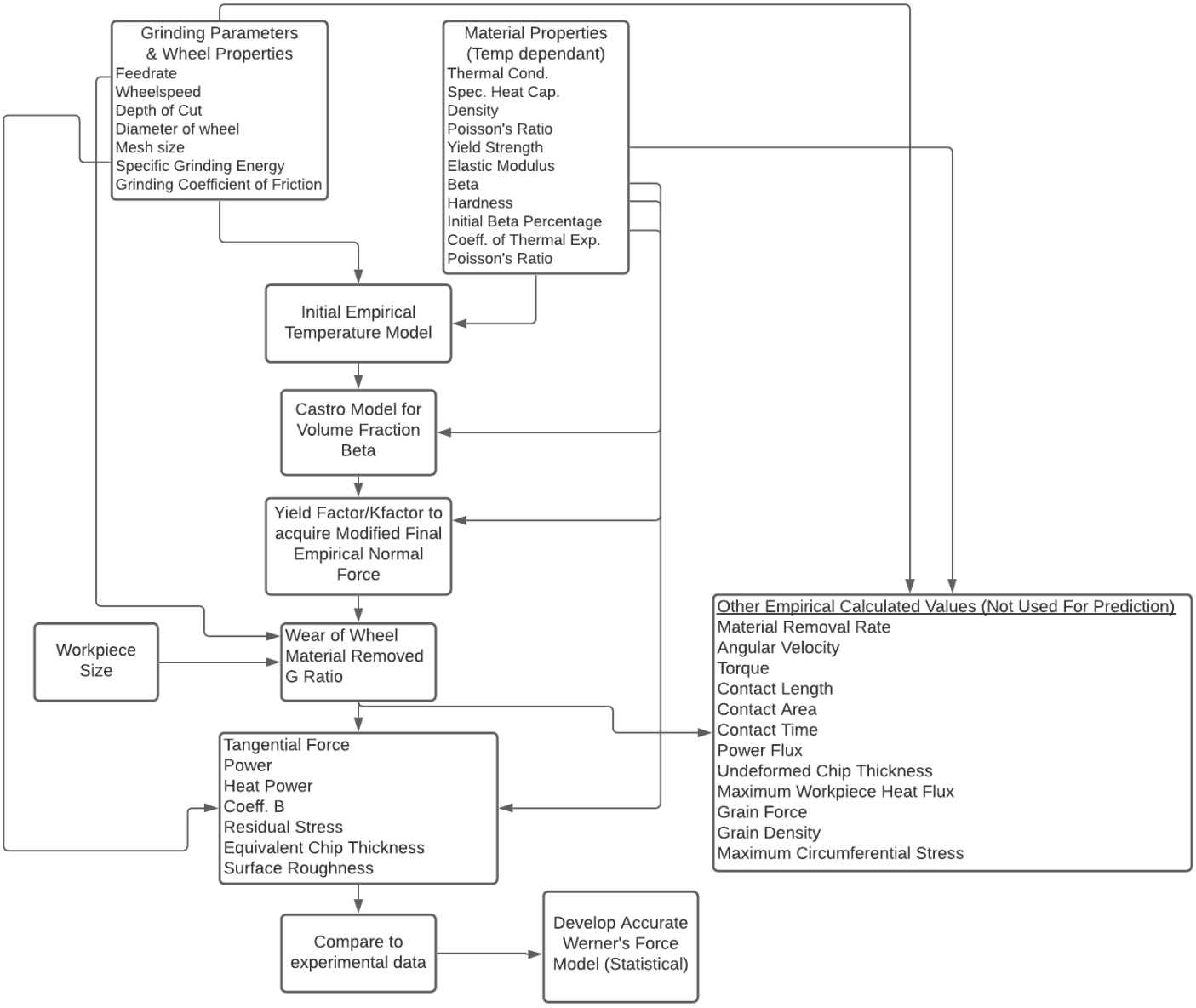

Figure 26 - Final Model Flowchart

The temperature change calculation for the final model was changed from the initial model as the updated model allows for calculation of temperature independent of heat flux into the workpiece, meaning that prediction of phase transformation and temperature varying properties can be completed without the need for multiple iterations (which would not be conducive to a live prediction environment).

The updated formula [41] for final temperature is given as:

$$T = 25°C + 1.13C_p \frac{v_w^{0.5} a_e^{0.75}}{\beta d_s^{0.25}} \tag{17}$$

Where all stated values are the same as those named in the initial model except for:

d_s = diameter of the wheel

The initial and final temperature models were compared to note any relationships between the two and it was found that there existed a linear correlation, with the final model showing temperatures only slightly higher than those of the initial model. The ambient temperature before grinding is assumed to be 25 degrees Celsius.

The model created is based on Ti-6Al-4V and thus the temperature dependent properties of this material were considered as well as its phase transformation characteristics, in order to create a general model that can consider a large variety of materials.

Phase transformations occur in certain materials and this affects the ability to machine the material (as noted in the literature review).

The Castro model [41] was used to calculate the final phase transformation percentage regarding the calculated maximum temperature. It should also be noted that workpieces are often used in hardened or annealed states with an initial phase transitional percentage based on previous metalworking processes. This was assumed to be 10% in the development of the final model but it is accounted as a variable in the system that can be changed based on material acquired.

The Castro model is given by:

$$f_\beta = 0.075 + 0.92e^{-0.0085(\beta_f - T)}, 298K < T < \beta_f \tag{18}$$
$$f_\beta = 1, T > \beta_f$$

Where

$f_\beta =$ Fraction beta phase (between 0 and 1)

$\beta_f =$ Beta transition temperature

$T =$ Temperature of workpiece

The associated alpha phase of the workpiece can be acquired by subtracting the fraction beta phase from 1 (100%). This fraction of transition can then be used to assess the state of the workpiece at the specified grinding conditions to account for temperature dependant properties that determine machinability and thus determine final output force.

It is then necessary to determine how hardness and yield strength change with regards to temperature in order to create a scaling factor for force variation. This is where previous experimental work done on temperature dependant material properties comes into play. The change in hardness can be accounted for by dividing the initial hardness state magnitude (in BHN) by the final (or current for a live system) hardness state magnitude (in BHN) based on how much phase transformation has occurred (as given above). The final hardness state is found by the following extract of code in the model:

```
alphahardness = 300; %fully alpha BHN
    betahardness = 379; %fully beta BHN
    Workhardness(n) = alphahardness + percentagebeta(n)*(betahardness-
alphahardness);
```

The hardness factor is given by:

$$H_{ratio} = \frac{Work_{Hardness}}{\alpha_{stateHardness}} \tag{19}$$

The yield factor can be found in a similar state by dividing the two transitional hardnesses and an appropriate force factor (known as k-factor in industry) can then be found as shown in the code below:

```
alphayield = 7.6;
betayield = 72.3;
yieldfactor = betayield/alphayield
hardfactor = (Workhardness/alphahardness)';
forcefact = ((yieldfactor-1)*percentagebeta + 1)';
```

This scaling (force) factor can then be multiplied into the output force empirical model (as stated in the initial model section) in order to acquire a final force value.

The wear of the wheel was calculated using equation (2), and thus the grinding ratio (G-ratio) was found. The G ratio is a measure of the grinding wheel's ability to remove material effectively from the workpiece. It is the ratio between volume of workpiece removed to volume of wheel removed and generally reaches a constant after a short wear-in time.

Finally, the slightly modified residual stress can be calculated as:

$$\sigma = \frac{E(z)\alpha(z)\Delta T}{(1-\nu)} * H_{ratio} \tag{20}$$

Where all variables have been stated previously.

This concludes the development of the semi-analytical/empirical model, which allows for creation of experimental 'mock' data that is needed for development of the feature correlation and optimization engine to take place (the other team member's scope). Creation of this data is discussed in the following section.

4.1.3 Experimental Mock Data Creation & Feature Correlation

In order to develop the feature correlation (prediction and optimization) engine, a large amount of data is required. This data is used to train and test the developed machine intelligence (in this case it is an artificial neural network). The data is acquired from the model discussed in the previous section and can be split into two major sections: inputs to the neural network, and required outputs from the neural network. Once the neural network is developed, an accurate estimation of outputs should be available for inputs that have higher varying properties. These data inputs and outputs would normally be acquired through many grinding experiments. This was not possible due to the pandemic occurring this year, however, the theoretical model developed provides results that are more than sufficient to complete the same tasks and allows for the benefit of efficient change of inputs, and creation of outputs.

Please note that the scope of the other team member on this project includes the feature correlation development of this system and thus the neural network formation and advances are not included in this report.

As discussed in the literature review and the previous section, the main purpose of the IGS is to minimize workpiece burn and to limit residual stress development. The Coefficient B is used as a measure of residual stress and the empirical temperature formulas are used to calculate maximum workpiece temperature. The calculated residual stress is also used as a comparison to the calculated coefficient B. The required outputs from the neural network are thus: temperature, coefficient B, and associated residual stress.

As a dynamometer is to be used and has been used in the grinding set-up, force is a definite input to the neural network. The other known properties of the system, which vary according from case-to-case are work-speed, wheel-speed and depth of cut. These are the other inputs into the neural network. The material and wheel properties as well as grinding specific properties (coefficient of friction and specific grinding energy for example) are not chosen to vary and rather vary as a function of the other inputs changing. Wheel type, workpiece material, coolant choice and dress data could be inputs into the neural network and if fixing three of the four, the remaining variable could also be an output to the neural network. This again was not developed as it would reach out of the scope of this final year project, however, functionality for it in the user interface and model has been included and it could be developed further.

Mock data was created for a range of wheel speeds and wheel speeds with the previously created model. Wheel speeds were sampled at 30 m/s, 45 m/s and 60 m/s and work speeds were sampled at 10000 mm/min, 12500 mm/min and 15000 mm/min. This resulted in 9 unique data sets. The depth

of cut was varied for each data set (with 30 depths of cut varying from 0 micrometres to 30 micrometres). The figures below show the achieved results for a 30 m/s and 10000 mm/min data set. Figure 27 below shows the normal grinding force vs depth of cut for the specified conditions while Figure 28 (on the following page) shows a simple response surface plot of the depth of cut vs residual stress vs depth of cut and Figure 29 shows a response surface plot for depth of cut vs surface temperature vs induced residual stress.

Table 1 (two pages forward) shows all the results of interest acquired from the model and the grinding ratio for the specified conditions is shown below the table. Finally Figure 30 (three pages forward) shows the residual stress vs. coefficient B for the specified conditions and Figure 31 shows the residual stress state vs depth of cut.

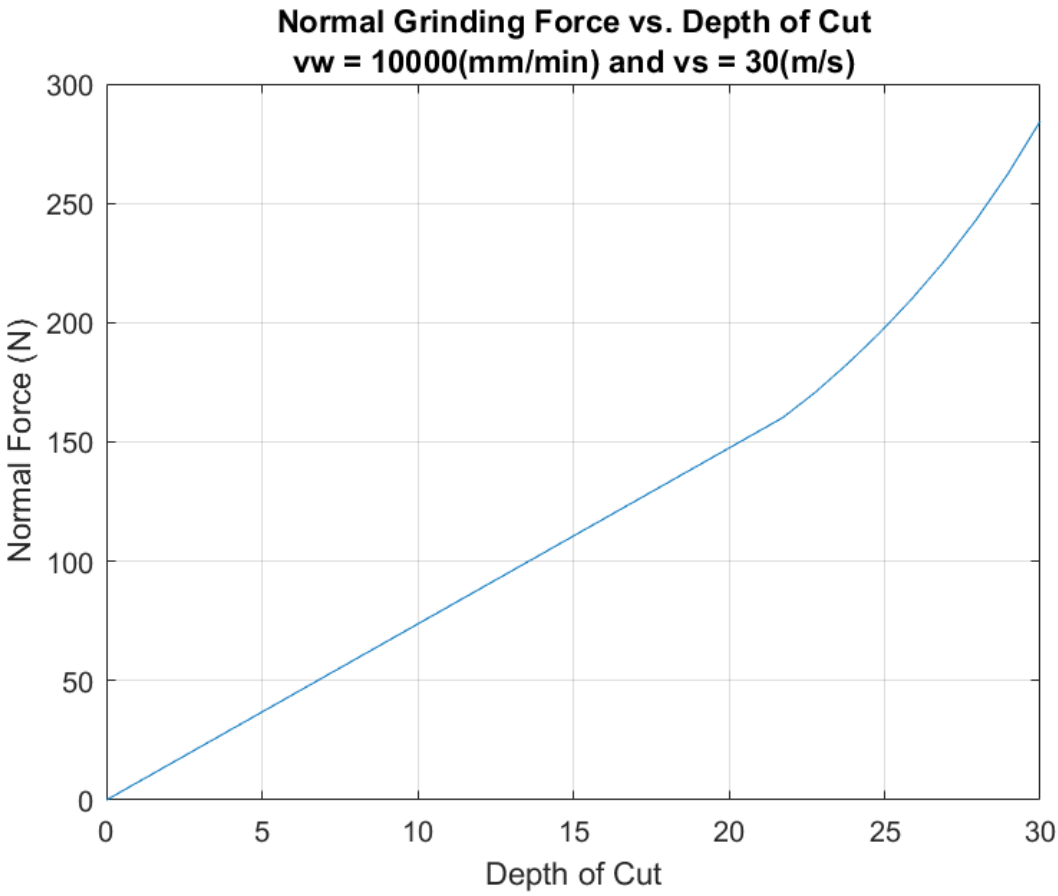

Figure 27 - Force vs. Depth of Cut Vw = 10000mm/min, Vs = 30m/s

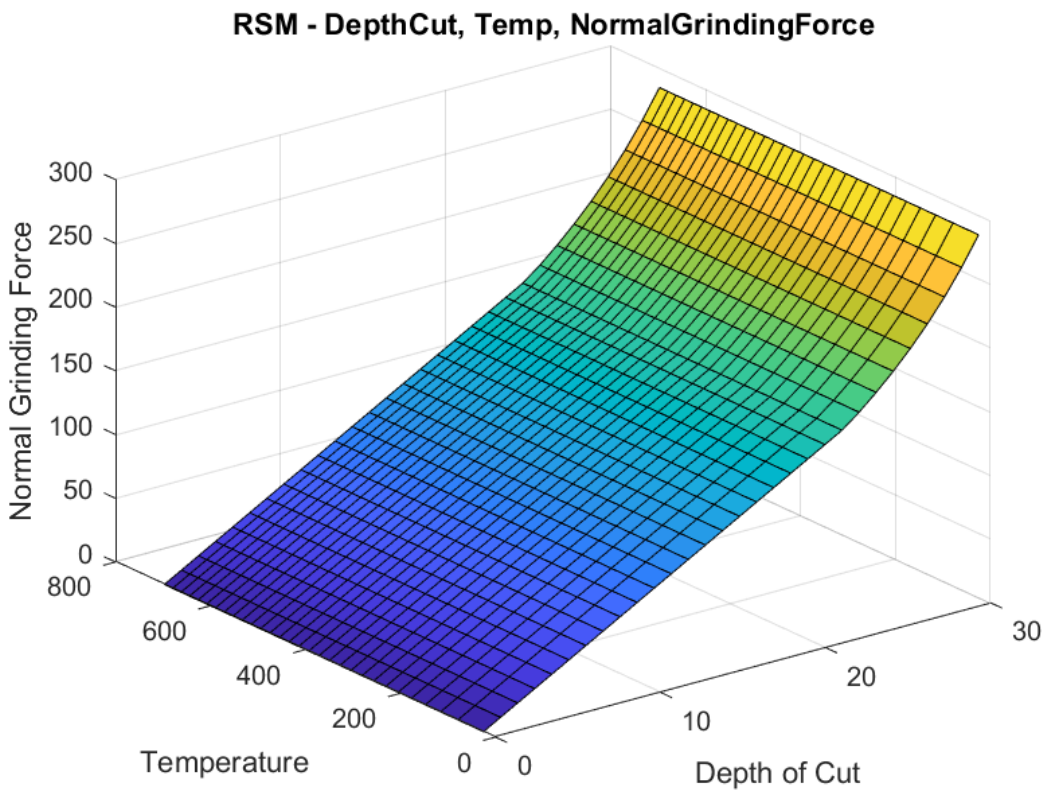

Figure 28 - Surface Plot Vw = 10000mm/min, Vs = 30m/s

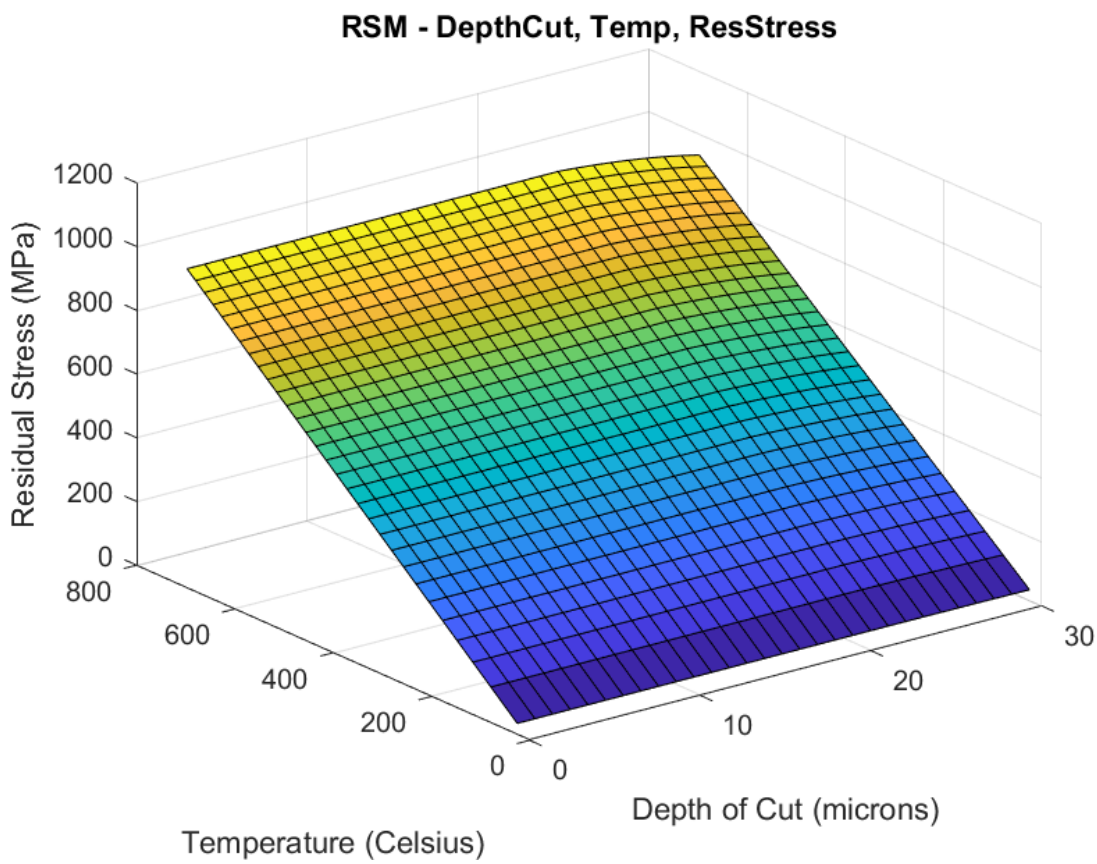

Figure 29 - Surface plot 2, Vw = 10000mm/min, Vs = 30m/s

	Depth of Cut (microns)	Normal Force (N)	Tangen Force (N)	Temperature (° C)	RStress (MPa)	Coeff. B	VolWorkRemoved (mm^3)	SurfRough
1	0	0	0	25.0000	0	0	0	Inf
2	1.0345	7.6262	3.3798	78.7843	88.0445	0.0304	0.0928	0.2744
3	2.0690	15.2524	6.7596	115.4540	148.0727	0.0608	0.1856	0.1372
4	3.1034	22.8786	10.1395	147.6017	200.6982	0.0913	0.2784	0.0915
5	4.1379	30.5049	13.5193	177.1249	249.0276	0.1217	0.3712	0.0686
6	5.1724	38.1311	16.8991	204.8386	294.3947	0.1521	0.4640	0.0549
7	6.2069	45.7573	20.2789	231.1906	337.5327	0.1825	0.5568	0.0457
8	7.2414	53.3835	23.6587	256.4616	378.9012	0.2129	0.6496	0.0392
9	8.2759	61.0097	27.0385	280.8426	418.8128	0.2433	0.7424	0.0343
10	9.3103	68.6359	30.4184	304.4714	457.4929	0.2738	0.8352	0.0305
11	10.3448	76.2621	33.7982	327.4513	495.1109	0.3042	0.9280	0.0274
12	11.3793	83.8884	37.1780	349.8628	531.7983	0.3346	1.0208	0.0249
13	12.4138	91.5146	40.5578	371.7699	567.6601	0.3650	1.1136	0.0229
14	13.4483	99.1408	43.9376	393.2247	602.7816	0.3954	1.2064	0.0211
15	14.4828	106.7670	47.3175	414.2705	637.2334	0.4259	1.2992	0.0196
16	15.5172	114.3932	50.6973	434.9434	671.0748	0.4563	1.3920	0.0183
17	16.5517	122.0194	54.0771	455.2743	704.3564	0.4867	1.4848	0.0171
18	17.5862	129.6456	57.4569	475.2898	737.1216	0.5171	1.5776	0.0161
19	18.6207	137.2719	60.8367	495.0129	769.4083	0.5475	1.6704	0.0152
20	19.6552	144.8981	64.2166	514.4638	801.2493	0.5779	1.7632	0.0144
21	20.6897	152.5243	67.5964	533.6604	832.6739	0.6084	1.8560	0.0137
22	21.7241	160.1505	70.9762	552.6183	863.7080	0.6388	1.9488	0.0131
23	22.7586	170.6524	75.2187	571.3519	895.1841	0.6770	2.0747	0.0125
24	23.7931	182.5635	79.8840	589.8736	926.6856	0.7190	2.2168	0.0119
25	24.8276	195.5284	84.8654	608.1951	958.1206	0.7638	2.3708	0.0114
26	25.8621	209.7440	90.2221	626.3266	989.5506	0.8120	2.5389	0.0110
27	26.8966	225.4423	96.0235	644.2777	1.0210e+03	0.8642	2.7237	0.0106
28	27.9310	242.8967	102.3518	662.0569	1.0527e+03	0.9212	2.9282	0.0102
29	28.9655	262.4292	109.3035	679.6723	1.0845e+03	0.9837	3.1556	0.0098
30	30.0000	284.4198	116.9926	697.1310	1.1167e+03	1.0529	3.4102	0.0095

Table 1 - Results of Model Vw = 10000mm/min, Vs = 30m/s

```
G = 20.3960     (Grinding Ratio)
```

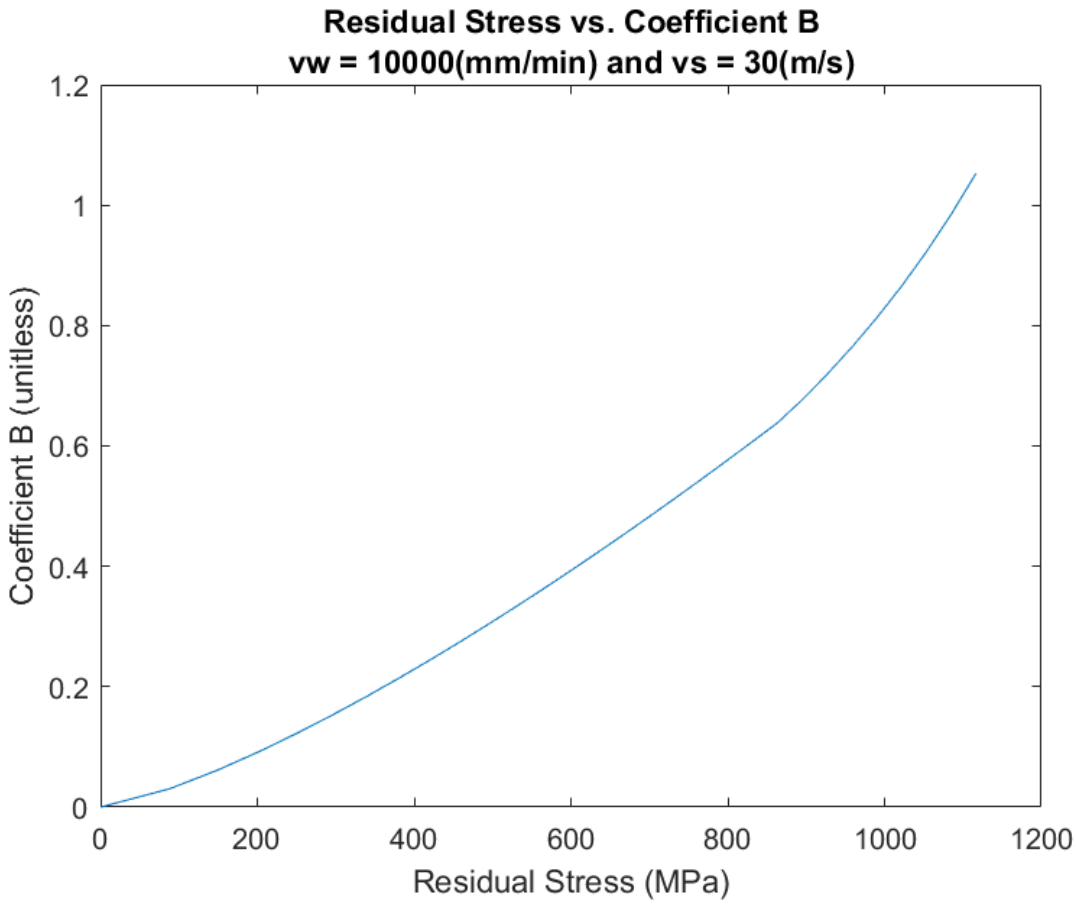

Figure 30 - Coefficient B vs. Residual Stress Vw = 10000mm/min, Vs = 30m/s

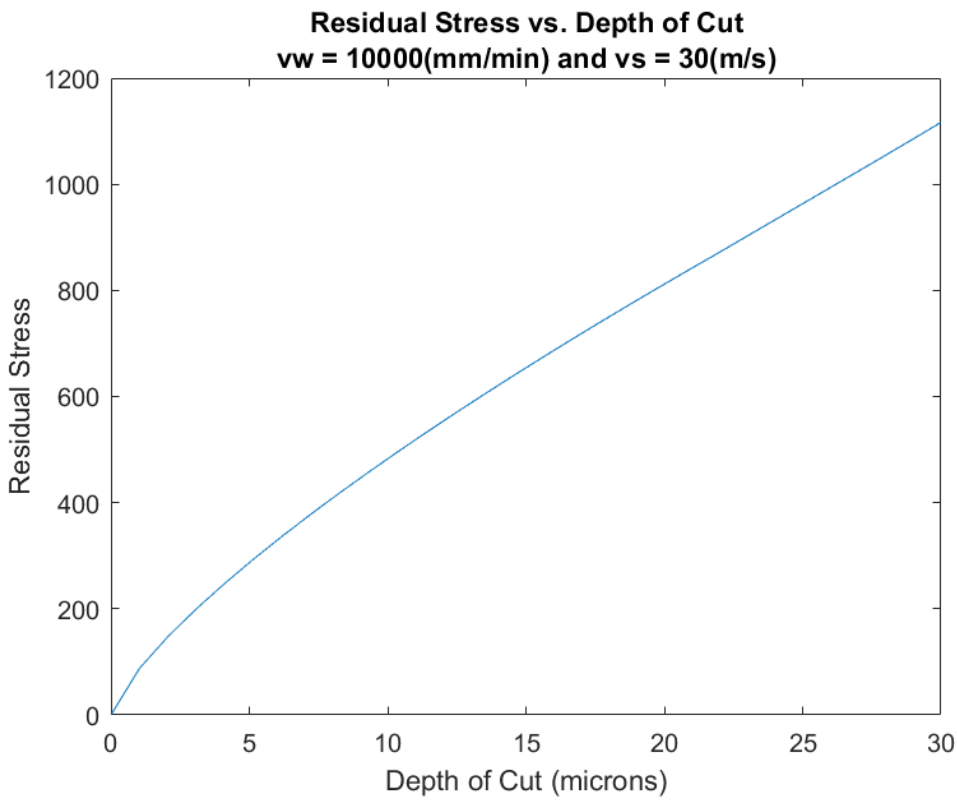

Figure 31 - Residual Stress vs. Depth of Cut - Vw = 10000mm/min, Vs = 30m/s

The results above are used extensively to create the feature correlation engine and the model is versatile in that the grinding conditions, wheel properties and work material can be changed. This is an integral feature that is included in the actual system and human-machine interface as well as the database.

The graphs above are made in this manner for effective analysis of aims and stipulated outputs. Note that a description and analysis of the results is discussed in the results section. To acquire results for any wheel speed or work speed, the model (attached in the Appendix) can be adjusted accordingly and run in MATLAB.

4.1.4 Past Experimental Data Use

Some experiments had been completed in 2017 by a previous UCT final year student. The work done was mostly done on varying coolants with most grinding conditions remaining the same. However, the data could still be used for feature correlation and the few experiments regarding varying grinding conditions do still contribute to helping to train and test a neural network. Figure 32 below shows a full grinding experiment for a 6-micrometre depth of cut, a grinding wheel speed of 29.74 m/s and a work speed (feedrate) of 13752 mm/min. This particular experiment took around 50 seconds to complete.

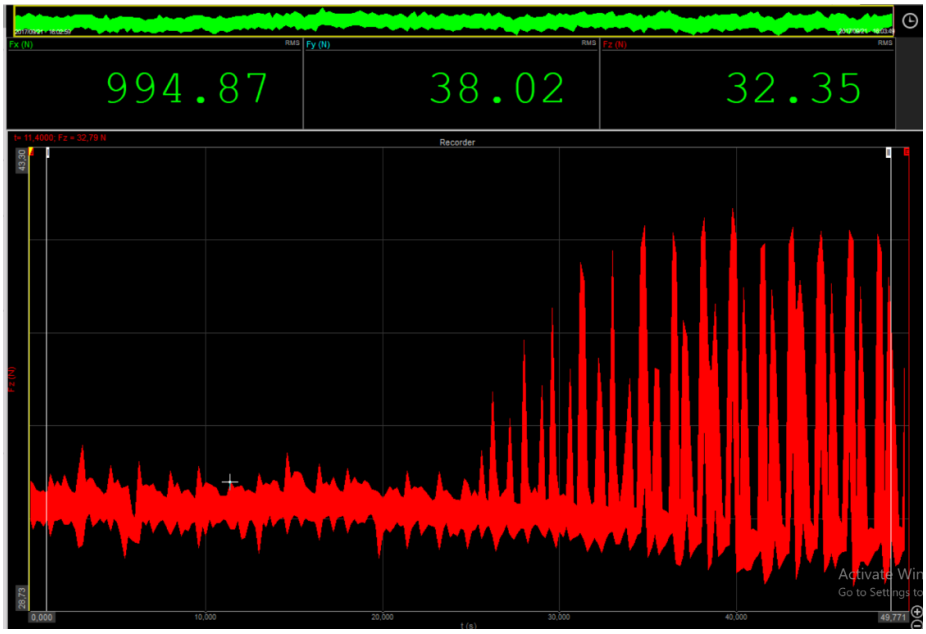

Figure 32 - Grinding Normal Force vs Time for a Full Experiment (DeweSoft)

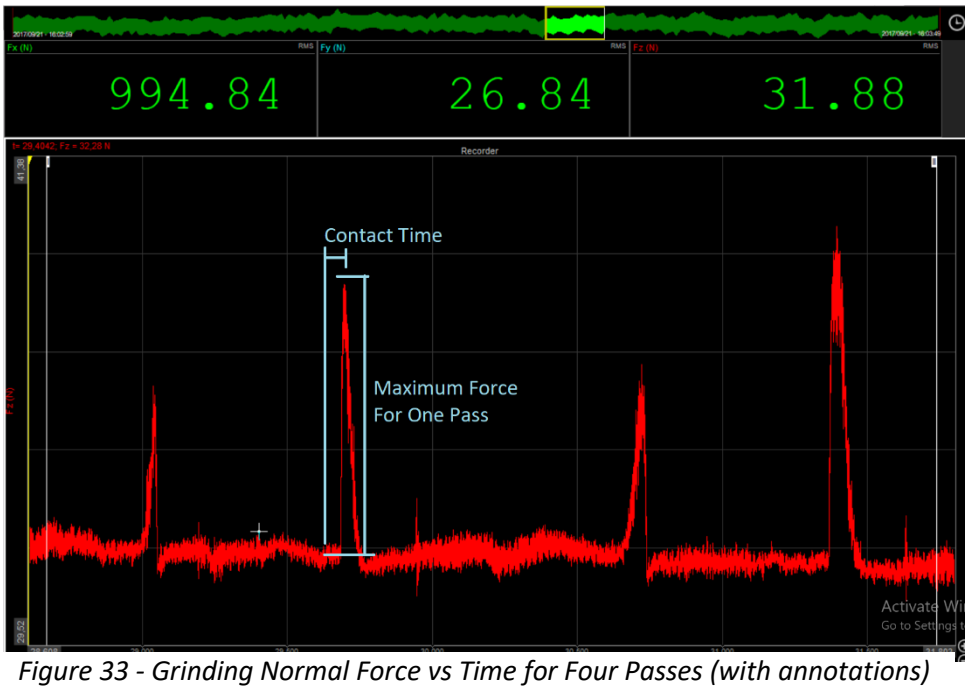

Figure 33 - Grinding Normal Force vs Time for Four Passes (with annotations)

Figure 33 on the previous page shows an analysis for a few passes (each spike is a contact with the wheel and workpiece). The contact times are around 0.1s and the maximum force varies a large amount. These two variables were measured for 10 passes per experiments at different points to find maximum and average forces as well as a large amount of analytical formulas (as described in the empirical model) used to extract features. The calibration factor for the dynamometer (actual force to measured force) was found to be 10 (as per the calibration files). This means that the measured forces needed to be multiplied by 10 to acquire the actual force. The results were summarised in a excel file with multiple tabs for 13 experiments. Figures 34 and 35 below show the corresponding excel file of extracted data for the DeweSoft data on the previous page. The residual stress column was left out as this was to be predicted with the neural network developed by the other team member.

	S (Gwheel RPM)	F (feedrate mm/min)	D (depth of cut) microns	Q Project	Number	EXP			Calibration Factor
	2840	13752	6	N/A	N/A	N/A			10
	velocity (wheel speed in m/s) = S*r	Feedrate (m/s)						le = lc	
	29.74041045	0.2292						$t_c = \frac{l_e}{v_w}$	Fn=DeltaFz*Calibration Factor
	S is rpm and r is radius of wheel								
	Fz1 (N)	T1 (s)	Fz2 (N)	T2 (s)	DeltaFz (N)	DeltaT (s)	Gradient (N/s)	Theor time (s)	Fn (N)
	31.69	27.9988	37.24	28.012	5.55	0.0132	420.4545455	0.004779429	55.5
	31.56	29.6812	38.26	29.6982	6.7	0.017	394.1176471		67
	31.47	30.6596	36.33	30.724	4.86	0.0644	75.46583851		48.6
	31.4	31.3634	39.68	31.3966	8.28	0.0332	249.3975904		82.8
	31.45	33.0508	40.05	33.0806	8.6	0.0298	288.590604		86
	31.34	36.4142	40.09	36.4962	8.75	0.082	106.7073171		87.5
	31.07	39.7444	41.38	39.8956	10.31	0.1512	68.18783069		103.1
	31.34	36.4142	40.61	36.5598	9.27	0.1456	63.66758242		92.7
	31.29	38.0718	41.09	38.2282	9.8	0.1564	62.65984655		98
	30.86	38.7844	38.38	38.8206	7.52	0.0362	207.7348066		75.2
	30.94	43.0474	40.79	43.2572	9.85	0.2098	46.94947569		98.5
	31	46.3672	40.62	46.4486	9.62	0.0814	118.1818182		96.2
AVERAGES	31.28416667		39.54333333		8.25916667	0.085016667	175.1762419		82.59166667

Figure 34 - Extracted Excel Experimental Data Part 1

		Cp=	9500000000							
		ae=	0.000006		$l_c = \sqrt{D * a_e}$	where D = 0.2m			$B = \frac{P}{b_d v_w}$	
		b=	0.02		0.001095445					
Mu(%)	30		$F_t = C_p \frac{V_w a_e b}{V_s} + \mu F_n$		$A_c = b * l_c$	assume b=0.02				
Mu	0.3									
	Ft (N)		Ft from above (N)	P=Ft*v (W)	2.19089E-05	PowerFlux=Fn/A (W/m^2)	Ft/Fn	Residual Stress (Mpa)	Coefficient B =	B in Ws/mm^2
	16.65		25.43562185	495.1778341		2533216.828	0.458299		108023.0877	0.108023088
	20.1		28.88562185	597.7822501		3058117.613	0.431129		130406.25	0.13040625
	14.58		23.36562185	433.6151844		2218276.358	0.480774		94593.19032	0.09459319
	24.84		33.62562185	738.7517957		3779285.647	0.406107		161158.7687	0.161158769
	25.8		34.58562185	767.3025897		3925344.995	0.402158		167387.1269	0.167387127
	26.25		35.03562185	780.6857744		3993810.315	0.400407		170306.6698	0.17030667
	30.93		39.71562185	919.8708953		4705849.64	0.385215		200669.9161	0.200669916
	27.81		36.59562185	827.0808147		4231156.757	0.394775		180427.7519	0.180427752
	29.4		38.18562185	874.3680673		4473067.553	0.389649		190743.4702	0.19074347
	22.56		31.34562185	670.9436598		3432394.694	0.41683		146366.4179	0.146366418
	29.55		38.33562185	878.8291289		4495889.326	0.389194		191716.6512	0.191716651
	28.86		37.64562185	858.3082457		4390909.169	0.391327		187240.0187	0.187240019
	24.7775		33.56312185	736.89302		3769776.575	0.412155		160753.2766	0.160753277

Figure 35 - Extracted Excel Experimental Data Part 2

4.1.5 Noise Elimination and Statistical Analysis

When acquiring raw signals, noise is a given. The elimination of this noise is important in order to analyse signals properly and to obtain meaningful outputs. The method used to eliminate noise in this system is that of discrete wavelet transformations and moving average filters. Figure 36 below shows a raw force vs. time signal acquired from the past experimental studies done by a previous student. These were exported from DeweSoft into a readable MATLAB format. Figure 37 below this shows the denoised signal that had a discrete wavelet transformation and a moving average filter applied to it. The loss of noise is very apparent (due to the wavelet transformation) and the smoothened line is due to the moving average filter.

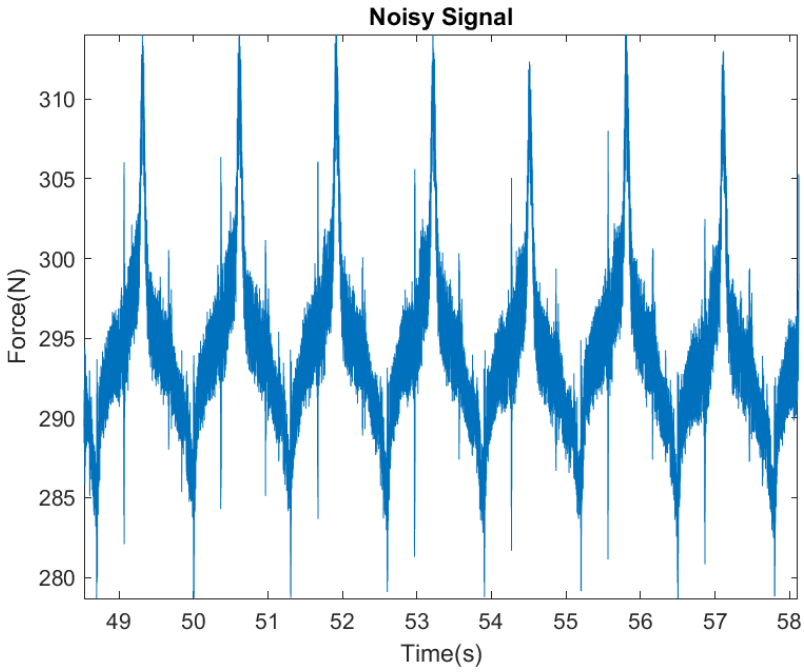

Figure 36 - Noisy Force Signal (imported into MATLAB)

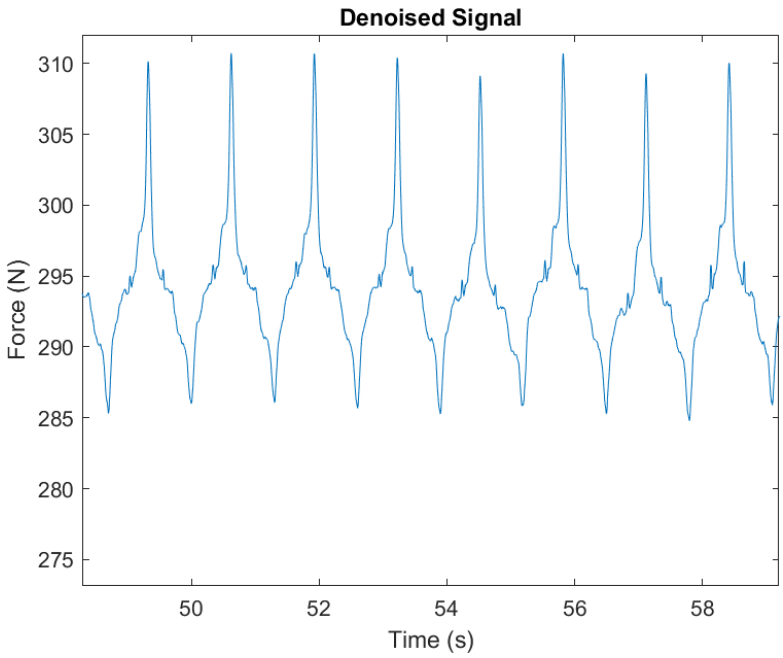

Figure 37 - Corresponding Denoised Signal

The MATLAB code for denoising of a sample signal can be found in the Appendix. The correct transformation was acquired using MATLAB's wavelet toolbox, which aids in suggesting a correct transform. A symmetric wavelet of order 2 was used with a universal threshold used as the data threshold. The universal threshold uses a fixed-form threshold. A soft threshold rule was followed (the standard rule) and the method of estimating noise in the data was defined as level independent. Level independent means to estimate noise variance based on highest-resolution wavelet coefficients.

The kurtosis coefficient is a great example of what can be read off the denoised signal and its value gives an indication of wheel wear. The kurtosis coefficient determines the sharpness/smoothness of a curve and when concerning force data, the greater the kurtosis coefficient, the sharper the curve, and inherently, the greater the wheel wear. There is a simple built in function in MATLAB that allows for one to acquire the kurtosis coefficient for a curve by giving the function denoised signal as an input. It returns a single value of kurtosis coefficient. When analysing the data, the wheel wear over each pass can be found by limiting the values over which data is taken, to that pass, and then sending that data to the function. Although this is not explored in more depth, it allows for a statistical model of wheel wear to be incorporated into the intelligent system.

No statistical modelling could be completed due to no laboratory access, however a simple model which determines the coefficients needed for Werner's force model (based on the empirical/semi-analytical model described in previous sections) can be developed. This opens the possibility of incorporating this model when live tests can be run again.

The main section for denoising by discrete wavelet transformations and moving average filters is given by:

```
fdenoise = wdenoise(double(forceset),5, ...
                'Wavelet', 'sym2', ...
                'DenoisingMethod', 'UniversalThreshold', ...
                'ThresholdRule', 'Soft', ...
                'NoiseEstimate', 'LevelIndependent');
coeff = ones(1,100)/100;
movavgforce = filter(coeff,1,fdenoise);
me = mean(movavgforce);
mov2 = movavgforce-me;
```

4.2 Database

The intelligent database was developed in Microsoft Access and contains all the relevant information for the grinding system to operate properly, for the developed model to acquire certain inputs, and for the system to store and reference outputs. The database structure was designed with the intention that the system would be used by a team or in the workplace, in the regard that it would not only allow for storage of process data and optimization of the process, but it would also keep track of part and batch numbers as well as operator usage, so that a proper referencing system can be implemented.

The database includes information regarding workpiece material properties, wheel material properties, active, optimised and reference cycle data as well as an individual part record. Coolant data is included as well, and its properties and connections have been properly defined. The dress data is not a part of the scope of this project but allowances in the database have once again been made for its inclusion if the project were to be taken further. Figure 38 below shows a flowchart of the developed database along with its external links to the intelligent system via human inputs and via system outputs. The material, wheel and coolant tables (as would the dress table if included) all link to the cycle data tables, which are then linked to the individual part record (IPR). The input by user is linked to both the IPR and the cycle data tables while the system outputs are linked to only the IPR.

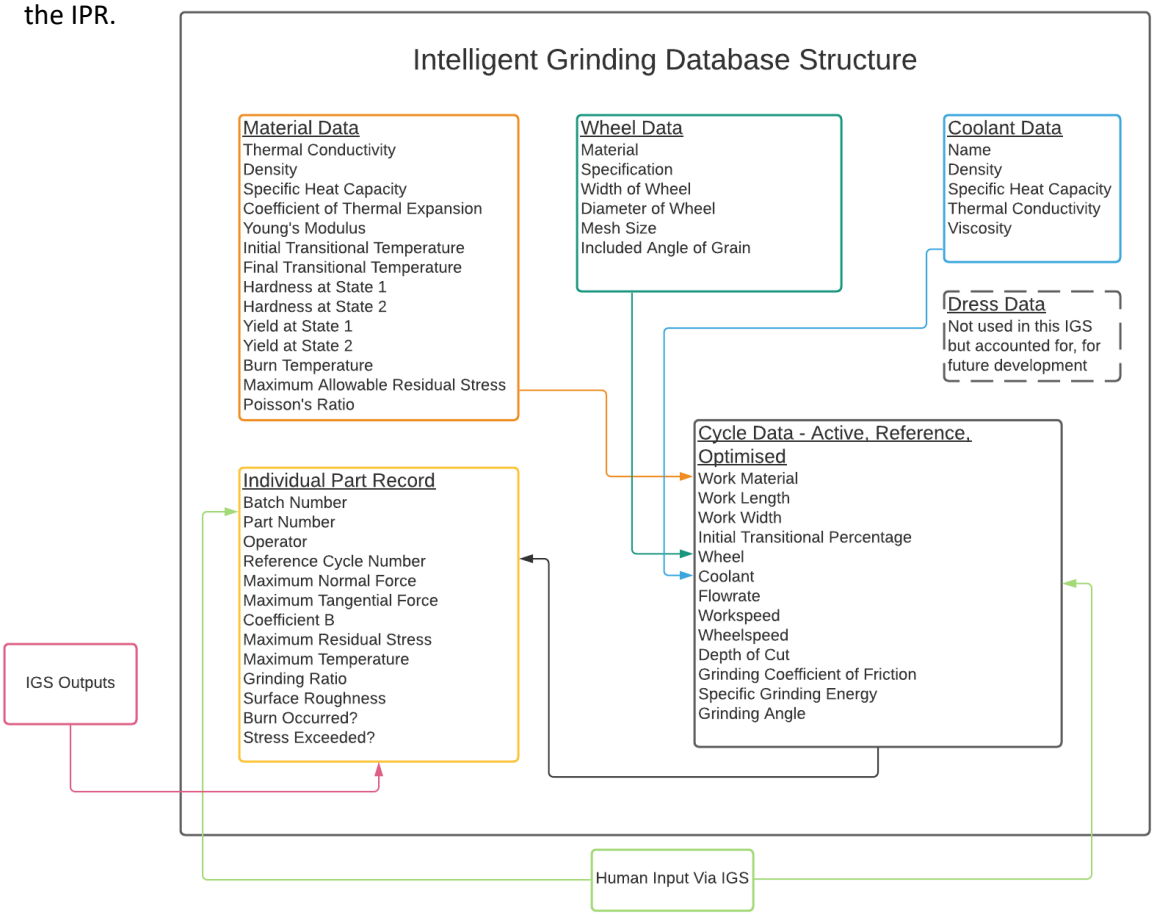

Figure 38 - Database Structure

A short description of each table and its' associated contents in the database will be described in the following paragraphs.

The material table contains the following constant material properties for each unique material: thermal conductivity, density, specific heat capacity, coefficient of thermal expansion, young's modulus and poison's ratio. The properties included which state the limits of the material are burn temperature and maximum allowable residual stress. The transitional properties are initial transitional temperature, final transitional temperature and the associated hardnesses and yield strengths. These transitional properties are the important aspects included in the model when considering material property change due to increase in temperature. These properties were all found using MatWeb's online database [42].

The wheel table contains unique properties for each wheel, and these are: specification, material, width, diameter, mesh size and included angle of grain. These are used in the developed model for the system and for reference of wheel used. The wheel properties would affect dress cycles and would be used in a wheel optimization model too.

The coolant table contains the name, density, thermal conductivity, specific heat capacity and the viscosity of the coolant. Although coolant properties are not used in the model, the properties described are sufficient to integrate the effects of the coolant into the system, as used in Morgan et al.'s system [4].

The cycle data is split into three tables: active, reference, and optimised. The active cycle data describes the current state and properties of the system, while reference cycle data describes all previous system runs and optimised cycle data describes system runs where workpiece burn conditions and overstressed conditions have not occurred, as specified by the user. The included information is workpiece material, workpiece length and width, initial transitional material percentage, wheel choice, coolant choice, coolant flow rate, work speed, wheel speed, depth of cut, friction coefficient, specific grinding energy and grinding angle. The cycle data describes all inputs to the system.

The final included table is that of IPR. This keeps record of every system in a similar way to that of the reference cycle data but differs in the major aspect that it stores outputs and tracks necessary part information. Each part record links to the appropriate reference cycle in the same system run. This ensures that inputs can be tracked to the relevant outputs. The properties stored in the IPR are: batch and part numbers, operator, reference cycle ID, maximum normal and tangential forces,

coefficient B and corresponding maximum residual stress, maximum surface temperature, grinding ratio, surface roughness, and the presence of burn condition and/or overstress condition.

Snapshots of portions of the material data table (with two additional materials added besides Ti-6Al-4V), reference cycle data table and individual part record table are shown in the Figures 39, 40 and 42 below, respectively. Figure 41 below shows all tables, the query needed to relate IPR to Reference Cycle and all the forms required to create the database GUI. The database GUI will be described in the following section.

MaterialData

ID	MaterialName	ThermalConductivity	Density	SpecificHeatCapacity	CTE	YoungsModulus	BeginTransitionTemp	EndTransitionTemp	HardnessOne	HardnessTwo
1	Ti-6Al-4V	6,7	4430	526,3	9,15	113,8	883	980	300	379
2	AISI E9310 Steel	51,9	7850	472	12,2	200	725	790	363	305
3	Aluminium 6061	167	2700	896	25,2	68,9	586	586	95	95
(New)		0	0	0	0	0	0	0	0	0

Figure 39 - Material Table

ReferenceCycleData

ID	WorkMaterial	WorkLength	WorkWidth	InitialTransPercent	Wheel	Coolant	Flowrate	Workspeed	Wheelspeed	DepthofCut	CoeffofFriction	SpecificGEnergy
1	Ti-6Al-4V	100	200	0,1	BZ100R100BQ1	Water	10	11000	30	9	0,32	9,5
2	Ti-6Al-4V	100	20	0,1	BZ100R100BQ1	Water	10	13000	20	9	0,3	9,5
3	Ti-6Al-4V	100	200	0,1	BZ100R100BQ1	Water	10	13000	20	6	0,34	9,65
4	Ti-6Al-4V	100	10	0,1	BZ100R100BQ1	Water	10	12500	30	9	0,3	9,5
5	Ti-6Al-4V	100	20	0,1	BZ100R100BQ1	Water	0	12500	30	6	0,3	9,75
6	Ti-6Al-4V	200	300	0,1	BZ100R100BQ1	Water	5	12500	30	9	0,3	9,3245
7	Ti-6Al-4V	100	200	0,1	BZ100R100BQ1	Water	10	13500	30	9	0,4	9,456
8	Ti-6Al-4V	100	260	0,1	BZ100R100BQ1	Water	0	12500	30	9	0,3	9,65
9	Ti-6Al-4V	100	200	0,1	BZ100R100BQ1	Water	50	12500	30	9	0,3	9,5
(New)		0	0	0			0	0	0	0	0	0

Figure 40 - Reference Cycle Table

ID	PartNo	BatchNo	Operator	ReferenceCycleNumber	MaxFn	MaxFt	MaxCoeffB	MaxResidStr	MaxTemp	GrindingRatio	SurfaceRoughness	BurnOccurred
2	0	0	Quintin	2	64,1742	37,7773	0	531153000	335,647	1456,2	0,0415256	No
3	1	1	Fungai	3	434,583	273,208	0,126096	407,979	257,81	1433,57	0,0508583	No
4	1	1	Quintin	4	20,5687	12,1081	0,174357	521,607	329,614	2271,68	0,0267218	No
5	2	2	Fungai	5	28,1466	16,569	0,119297	404,568	255,655	2213,43	0,0327274	No
6	3	3	Ramesh	6	605,66	356,533	0,171136	512,702	323,987	2314,44	0,0267218	No
7	5	5	Kalvin	7	442,225	304,546	0,203031	538,197	340,098	2113,19	0,0288436	No
8	0	0	Quintin	8	543,229	319,781	0,17711	529,218	334,424	2236,37	0,0267218	No
9	3	3	Quintin	9	411,373	242,162	0,174357	521,607	329,614	2271,68	0,0267218	No
(New)	0	0		0	0	0	0	0	0	0	0	

Figure 42 - IPR Table

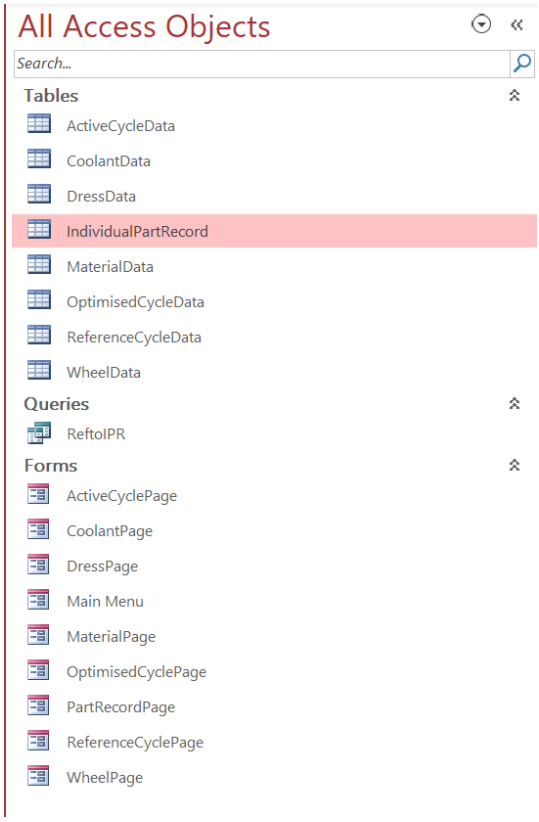

Figure 41 - All Tables, Queries and Forms of Database

4.3 Intelligent System Development and Human Machine Interfaces (Graphical User Interfaces)

The development of the final intelligent system involves the integration of the developed model, the created database, the statistical methods, the denoising methods, the human inputs, the system outputs, the acquired grinding force and time signals, the database GUI, the system (MATLAB) GUI and the DeweSoft (real-time analysis) GUI. Feature correlation, namely prediction (neural network) and optimization, would be integrated into the overall system at this stage as well.

Development of the database GUI, MATLAB GUI and DeweSoft GUI are discussed before describing the full integration in the results and discussion section that follows.

The database GUI allows for users to effectively view, input, edit and delete records. It also allows for simple and understandable navigation of the database. This was developed using Microsoft Access Forms in the same database as described in the previous section and was made to open as a user interface that is seemingly separate to the database (with some simple coding and access tricks). The database still opens in a minimized tab but no actual interaction with the tables is necessary to interact with the database. The database can still be opened if a user wishes to rather interact with the database tables.

Figure 43 below shows the main menu of the database GUI, which allows navigation to any of the table forms. Figure 44 on the following page shows the material form of the GUI as an example of what the GUI enables a user to do. All other tabs are of a similar format to the material form and have the same functionality imbedded into them.

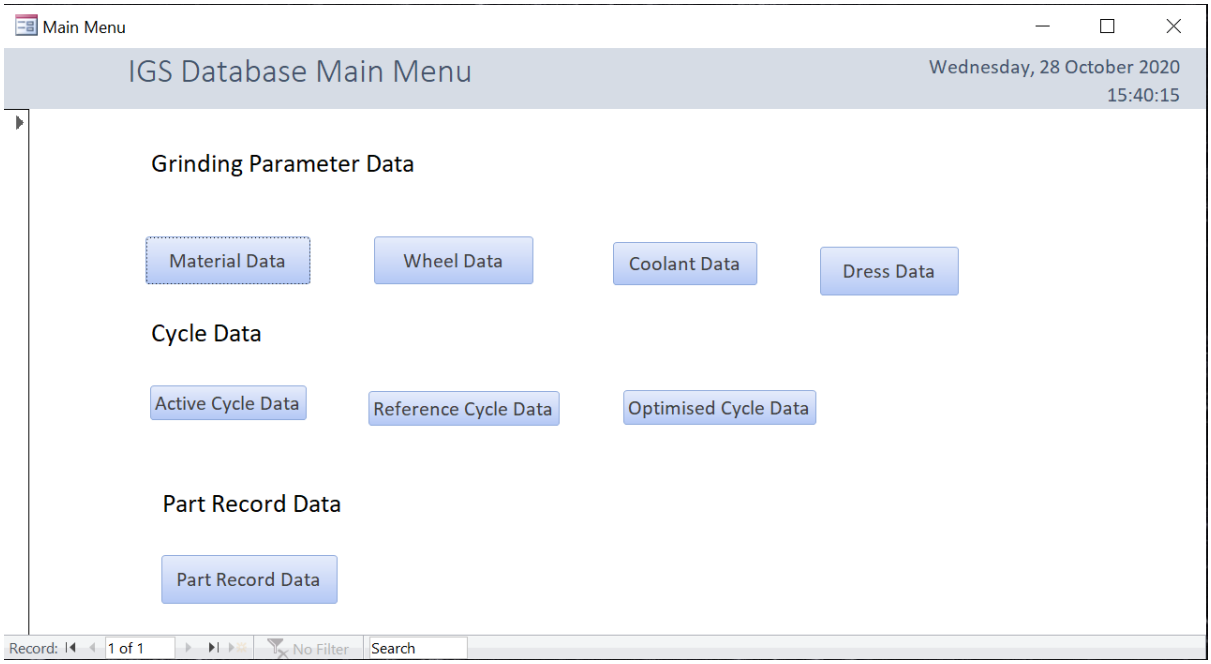

Figure 43 - Database GUI Main Menu

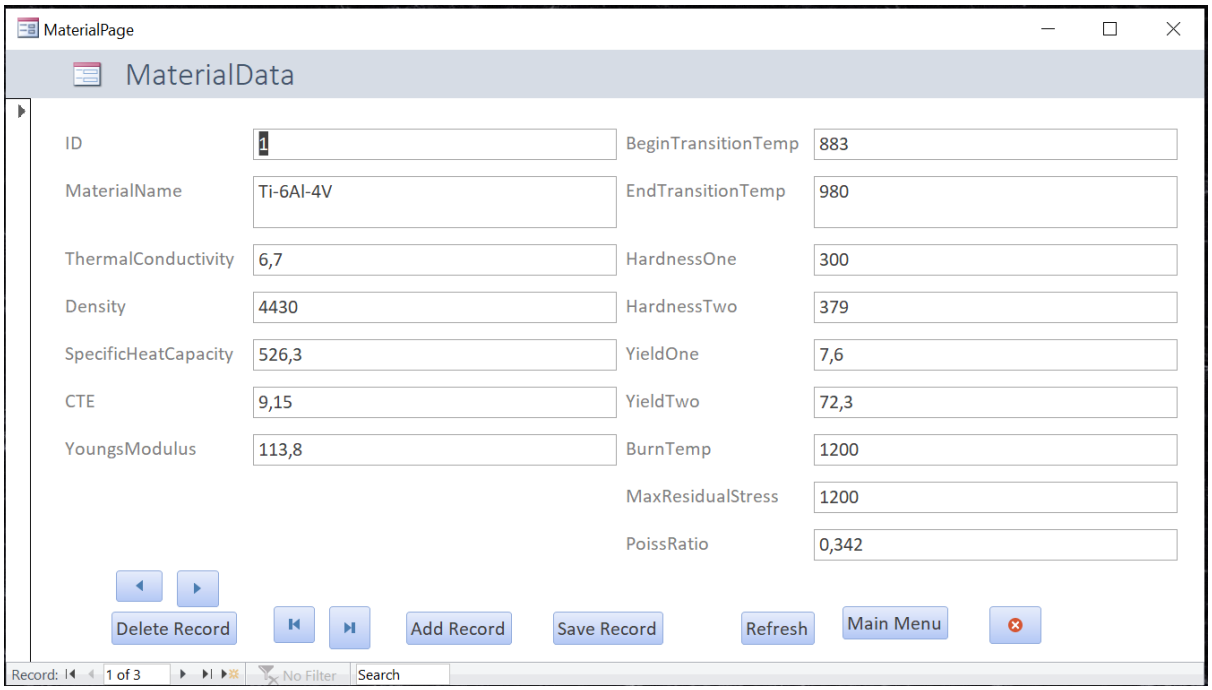

Figure 44 - Material Table GUI

The next developed human-machine interface was that of the system GUI. This was developed on MATLAB's App Designer feature and by use of code (MATLAB's own language, similar to the C programming language). The developed MATLAB GUI contains the core of this project and is also the heart of the integration between the database, user, and DeweSoft run page. The model developed is also in the backend of this GUI, along with the required connections to the database. The results and discussion section that follows describes the logic and integration of the MATLAB system and thus this methodology section will be kept short. The pages were designed and developed using a combination of design view and code view in the App Designer feature. Components were added as required (buttons, text, input fields, drop down menus, graphs, check boxes and tabs) in design view, as can be seen in Figure 45 on the following page, which shows the design of the output page.

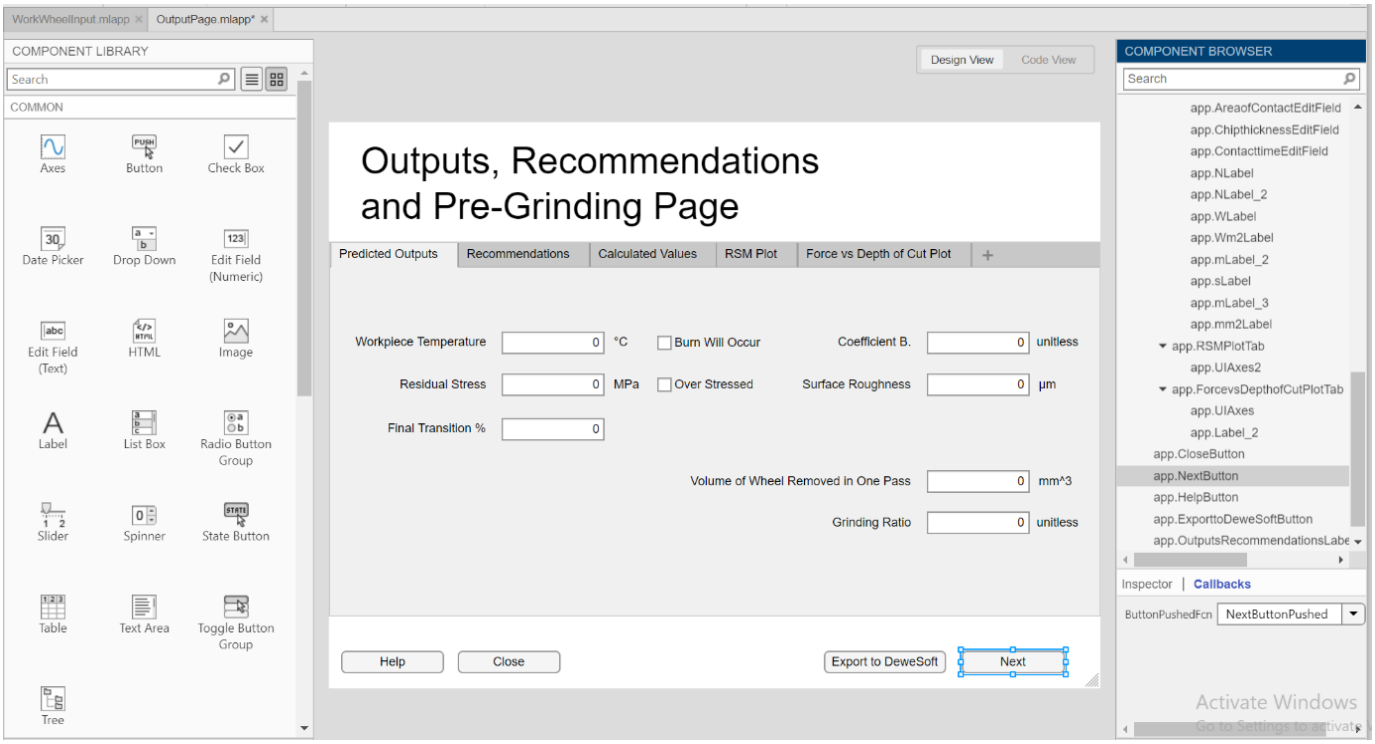

Figure 45 - Output and Recommendations App Design

The figures on the following two pages show a small section of the code view relating to the output page. These are the codes programmed for the start-up of the app and for the close button, help button and next button, respectfully. Most lines of code call either predefined methods e.g. closereq(), which closes the current page, or methods coded by myself which do a variety of functions e.g. defineproperties(app), which sends the information stored in the variable app, to the method definefunctions(), which then employs the model in code format and updates the output page to display correct values. Functions such as updateactive(app) and insertpartrecord(app), send information to the database and insert data to the active cycle and individual part record tables, respectively.

```
% Callbacks that handle component events
methods (Access = private)

    % Code that executes after component creation
    function startupFcn(app, startpage, wheelworkcoolant, gparams)
        app.startpage = startpage;
        app.wheelworkcoolant = wheelworkcoolant;
        app.gparams = gparams;

        defineproperties(app);
        updateactive(app);
        disp(app.temp);
    end

    % Button pushed function: CloseButton
    function CloseButtonPushed(app, event)
        closereq();
    end

    % Button pushed function: HelpButton
    function HelpButtonPushed(app, event)
        message = {'This is the Output and Recommendations Page';
                   'Click Open Dewesoft and all inputs will be sent there as well a
                   'Once Done in DeweSoft, click the export button in Dewesoft, the
        msgbox(message, 'Help', 'help');
    end

    % Button pushed function: NextButton
    function NextButtonPushed(app, event)
        insertreference(app);
        insertpartrecord(app);
        AnalysisPage(app.startpage, app.wheelworkcoolant, app.gparams);
        closereq();
    end
```

Figure 46 - Output Page Portion of Methods Code

An extract of the defineproperties() method can be seen in Figure 47 and the insert reference and insert part record methods can be seen in Figure 48. All relevant code for each section can be found in the Appendix.

```
depthcut = (app.gparams{1,3})*10^-6; %convert to m
desiredroughness = (app.gparams{1,4})*10^-6; %convert to m
coeffriction = (app.gparams{1,5}); %already in needed untis
spgenergy = (app.gparams{1,6})*10^9; %convert to J/m^3
gangle = deg2rad(app.gparams{1,7}); %convert to radians

Q = depthcut*workspeed*widthofcut;
dg = 15.2/meshsize;
C = 1/(2.25*(dg^2));
r = 10;

tana = tan(grainangle);

beta = sqrt(thermcon*density*spheat);
thetathree = 25+(1.13*spgenergy*(workspeed^0.5)*(depthcut.^0.75))/(beta*diameter^0.25);
fbeta = 0;
workhardness = 0;
pbeta = transper;

if ((25<thetathree) && (thetathree<= endtrans))
    fbeta = 0.075 + 0.92*exp(-0.0085*(endtrans-thetathree));
elseif thetathree> endtrans
    fbeta = 1;
end

if fbeta > pbeta
    pbeta = fbeta;
end
workhardness = hardnessone + pbeta*(hardnesstwo-hardnessone);

yieldfact = yieldone/yieldtwo;
hardfact = workhardness/hardnessone;

forcefact = ((yieldfact-1)*pbeta + 1)';
Fn = depthcut*forcefact*hardfact*spgenergy*((pi*workspeed*width)/(2*wheelspeed))*tana;
Ft = spgenergy*(workspeed*depthcut*width/wheelspeed) + coeffriction*Fn;
P = Ft*wheelspeed;
Pw = 0.65*P;
```

Figure 47 - Methods Code Portion 2

Design View | Code View

```
    end
    function insertreference(app)
        conn2 = database('IntelligentDatabase','','');
        workmat = convertCharsToStrings(app.wheelworkcoolant{1,2});
        length = (app.wheelworkcoolant{1,3});
        width = (app.wheelworkcoolant{1,4});
        transper = (app.wheelworkcoolant{1,5})/100; %convert
        wheel = convertCharsToStrings(app.wheelworkcoolant{1,1});
        coolant = convertCharsToStrings(app.wheelworkcoolant{1,6});
        flowrate = app.wheelworkcoolant{1,7}; %already in SI

        workspeed = (app.gparams{1,1});
        wheelspeed = (app.gparams{1,2});
        depthcut = (app.gparams{1,3});
        coeffriction = (app.gparams{1,5}); %already in needed untis
        spgenergy = (app.gparams{1,6}); %convert to J/m^3
        gangle = (app.gparams{1,7});
        datain = table(workmat,length,width,transper,wheel,coolant,flowrate,workspeed
          'VariableNames',{'WorkMaterial','WorkLength','WorkWidth','InitialTransPerce
        sqlwrite(conn2,'ReferenceCycleData',datain);
    end
    function insertpartrecord(app)
        conn2 = database('IntelligentDatabase','','');
        batch = app.startpage{1,1}; %integer
        part = app.startpage{1,2}; %integer
        operator = convertCharsToStrings(app.startpage{1,3}); %name

        sqlf = "SELECT MAX(ID) FROM ReferenceCycleData";

        reft = select(conn2,sqlf);
        app.refno = reft.Expr1000;
        datain = table(batch,part,operator,app.refno,app.fn,app.ft,app.coeffb,app.res
         'VariableNames',{'PartNo','BatchNo','Operator','ReferenceCycleNumber','MaxFr
        sqlwrite(conn2,'IndividualPartRecord',datain);
    end
```

Figure 48 - Methods Code Portion 3

The DeweSoft GUI is where the live grinding process occurs. The development of this portion of the human to machine interface is relatively simple in comparison to the MATLAB applications developed. Three channels are set up to measure force in three directions, where the z-direction measured force is the force used in analysis (normal force) and the only force considered. DeweSoft allows for math to be completed on the live signals and thus the model is brought in to calculate live coefficient B (thus residual stress) and temperature. A single channel is used to acquire data from the text file and subsequently display it in the run-page. The data is then exported to a file inside the system folder using the export button and choosing export type as 'Matlab (*.mat)'. This concludes the DeweSoft development methodology. Some of the channel set ups can be seen in Figure 49 below.

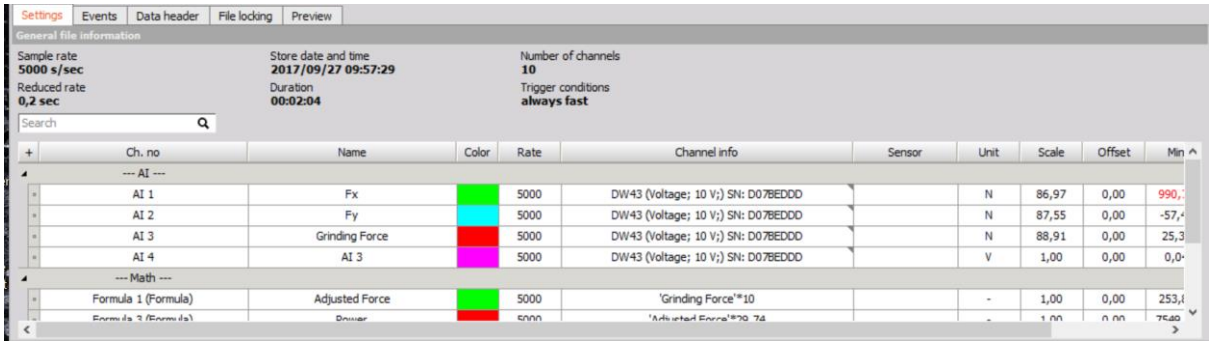

Figure 49 - DeweSoft Channels

Figure 50 below shows the export method with correct selections (text file data not included in the export).

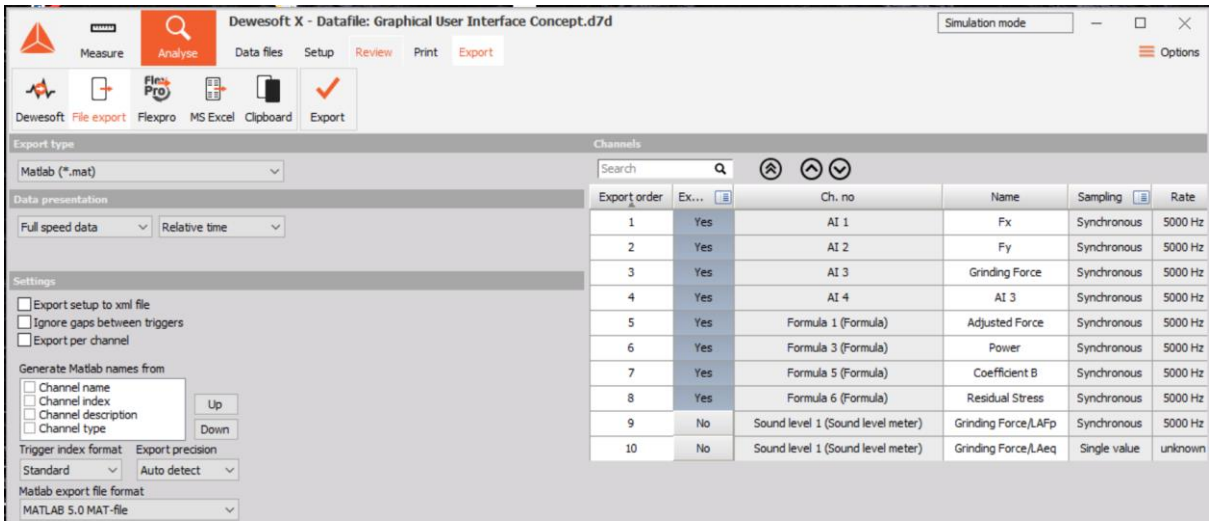

Figure 50 - DeweSoft Export Instructions

Figure 51 below shows the simple IGS run page. Note that this is for a previously stored data file and so the running of the system is not live. A set up file that works for live runs has been created as well but as there was no laboratory access or access to hardware components, no relevant or useful data could be created and so this previous data gives a more realistic example of what would be done in practice/industry. The example data file is that used in the past experimental data section described earlier.

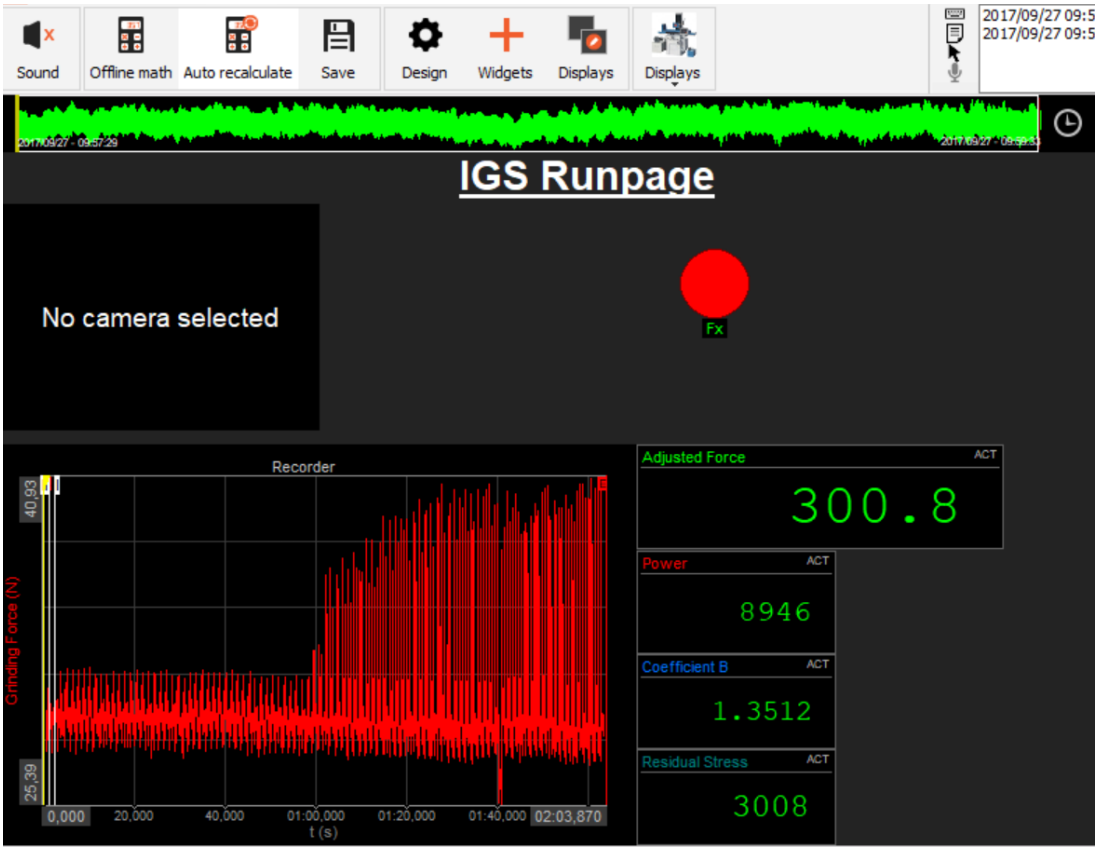

Figure 51 - DeweSoft Run page GUI

5 Results and Discussion

5.1 Full Integration of the System

Figure 52 below shows a flowchart of the full integration of all features of the system, including: database, live-systems, developed model, user required inputs, displays, outputs, database communications, live and predicted system communications, noise elimination, statistical analysis and where the neural network would be implemented if part of the scope. A full description of beginning to end of the system is discussed below the figure.

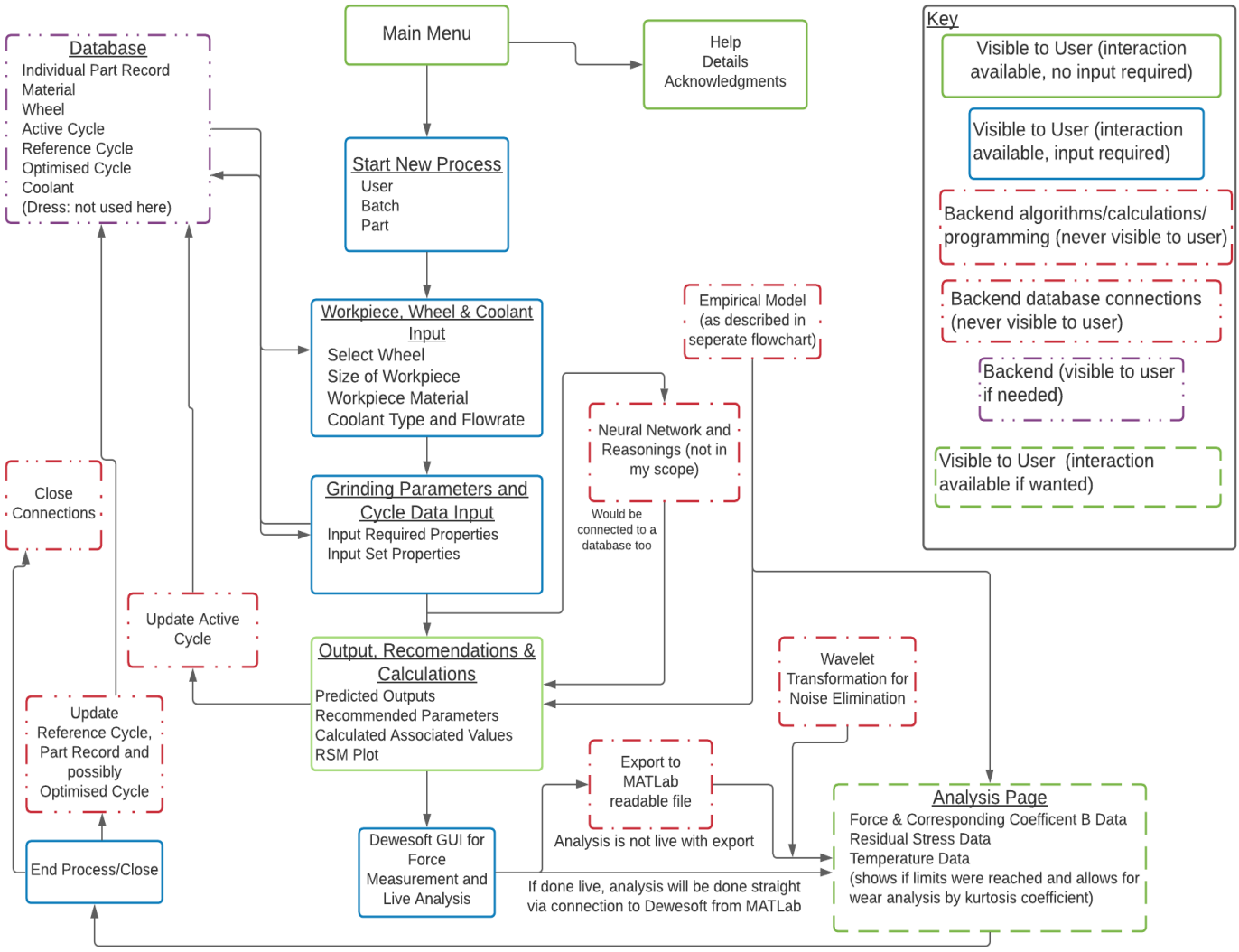

Figure 52 - Full System Integration Flowchart

The first display window when the program is run is that of the main menu. This allows an introductory step before beginning the process and gives the operator a chance to engage with the 'Guide to Use' manual in order to learn how to operate the system. The help, details and acknowledgment buttons display pop-ups with relevant information pertaining to their respective names. Note that each window/page has a help button that has information on what to do on the page, in order to clear any confusion a user might have.

The 'Guide to Use' describes, in detail, the steps the user should take to operate the system from beginning to end and can be accessed clicking on the programmed 'Guide to Use' button on the main menu window.

The primary button ('Start Process') on the main page takes the user to the starting page, where the part number and batch number are to be input, and an operator selected. This information is then taken (backend) to the 'Workpiece, Wheel and Coolant Input' page when the users clicks the programmed 'Next' button. The wheel choice and properties, material choice and dimensions, and coolant choice and flow rate are then to be input by the user. The wheel, material and coolant choices are linked to the associated database tables. The 'Next' button must then be pushed, and all previous inputted selections and data are carried forward to the following page ('Grinding Parameters and Cycle Data Input' page). The user must then input the required properties of: work speed (federate), wheel speed, depth of cut, desired surface roughness, optimization selection, specific grinding energy, and grinding angle. The desired surface roughness can be left at 0 if nothing is to be optimized for and a selection of none can be chosen if the user would not like to optimize anything. A button has been programmed, named 'Typical Values' that opens a window displaying the typical values for various grinding processes, so that the user can efficiently input data.

After the user presses 'Next' again, a multitude of different backend tasks occur while the user is taken to the 'Output, Recommendations and Pre-Grinding Page':

- All necessary data is acquired from the database pertaining to the material, wheel and coolant selected.
- The information carried throughout the system is taken to the 'Output, Recommendations and Pre-Grinding Page'.
- The developed model is run with the inputs from the user as well as the stored database information. All calculations are then completed in the backend and the output page is updated to display the relevant information.
- The Active Cycle table in the database is updated to contain only the information regarding the current cycle.

- If the feature correlation engine had been integrated, optimization would have occurred, and recommendations would have been duly updated as well.

If the user is satisfied with the predictions and outputs, they may continue to the next step. If not, they may exit the program and either: cancel the process or start the process again with new inputs.

The next step involves the user clicking the 'Export to DeweSoft' button. This exports the relevant outputs from the system to a text file, which can be read by DeweSoft as a channel to read from. The maximum allowable coefficient B and surface temperature is stated on the page clearly and the live coefficient B and surface temperature is stated beneath them, allowing the user to stop the process if the limits are close to being reached. Once the process is done, the export button must then be pressed, where the force and time data is export to a file in a specified location within the application folder. DeweSoft must then be exited and the user must return to the MATLAB system application ('Output, Recommendations and Pre-Grinding Page'). The 'Next' button is then pressed and this takes the user to the 'Analysis of Data' page.

The normal force data (modified to fit calibration) and time data is imported into the system and the noise is eliminated by the methods described earlier. The data is then displayed on a graph and the user can inspect the data accordingly. Functionality for the user to evaluate the kurtosis coefficient has been added and the associated wheel wear is displayed for the selected time period. The coefficient B vs. calculated residual stress is displayed in another graph and temperature vs. time is displayed in another graph. Note that the model described earlier is brought into this page too in order to display the associated information.

Once the user has completed analysis, they can save the cycle as optimal by clicking a button (feature correlation would enable this to be a built-in feature if part of the scope). If this button is clicked the database will be updated.

Finally, the user clicks the close button and the reference cycle table and individual part record table in the database are updated accordingly. The system has run its full course at this stage.

5.2 Developed IGS Package & Console

The IGS was planned to be developed for use in the University or to be marketed to a workplace. This means that some concept of package was required to be developed even if all work done was theoretical in nature. The figure below shows all necessary hardware required for the package along with corresponding software that is needed for the system to be installed. The most important hardware component for measurement of signals is the 9257 (Kistler) dynamometer, which measures torque signals from the grinding wheel into the workpiece. Amplifier and convertors are required in order to translate the measured signals into force signals. The DeweSoft DAQ is the hardware component required to connect software to hardware. The signals are displayed digitally here and stored appropriately. The operator has forms of input to both the MATLAB GUI and the DeweSoft GUI, with the operator interacting (not simply viewing) with the input pages and analysis page of the developed MATLAB GUI. DeweSoft interaction is in the form of starting the process, stopping the process (as required) and exporting the data to a MATLAB format. The algorithms and database communicate with each other (as mentioned in previous sections). This package can then be modified as necessary per customer requirements and a full and adaptable system can be marketed.

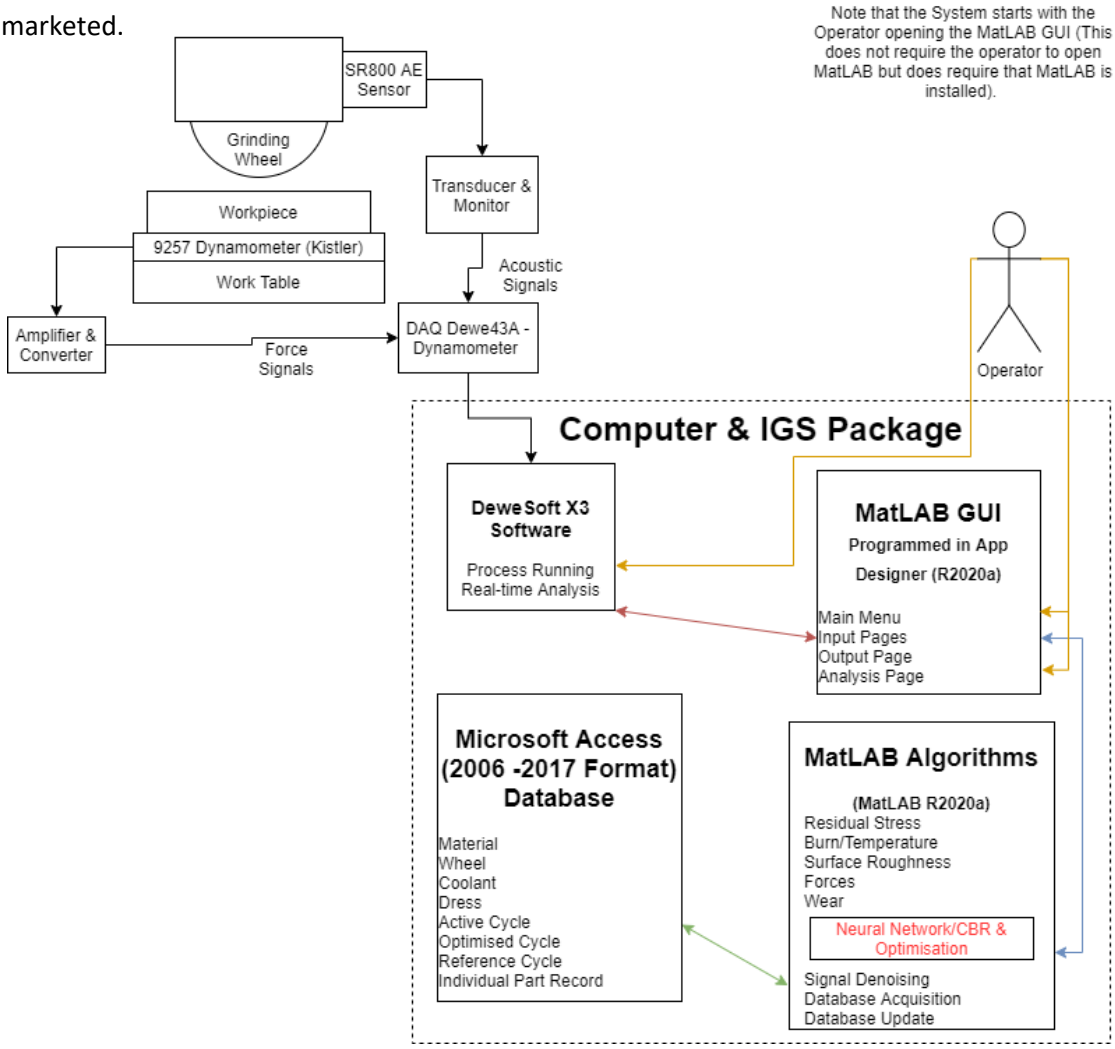

Figure 53 - Console and Package

5.3 MATLAB Model Developed Results and Discussion

A short analysis of the developed model results will be discussed here for three realistic cases with a lower work-speed and higher wheel-speed, a medium work-speed and medium wheel-speed and a higher work-speed and lower wheel-speed, respectively.

The results discussed in this section are specific to the model developed and the Ti-6Al-4V Alloy workpiece (initial phase transformation of 10%) ground with a 200mm diameter, 20mm wide, mesh size 100, Resin Bond Diamond grinding wheel. Work speed and wheel speed are kept constant and depth of cut vary for each case.

For a wheel speed of 60 m/s and a work speed of 10000 mm/min, a plotted graph of normal force vs depth of cut is shown in Figure 54 below. The force increases linearly as described by Equation 3, until phase transition begins. This happens at around 550 degrees Celsius where the phase-transus begins. The growth of force required then becomes exponential as the hardness of the material and yield strength are increased. This agrees with the nature of Ti-6Al-4V's material properties.

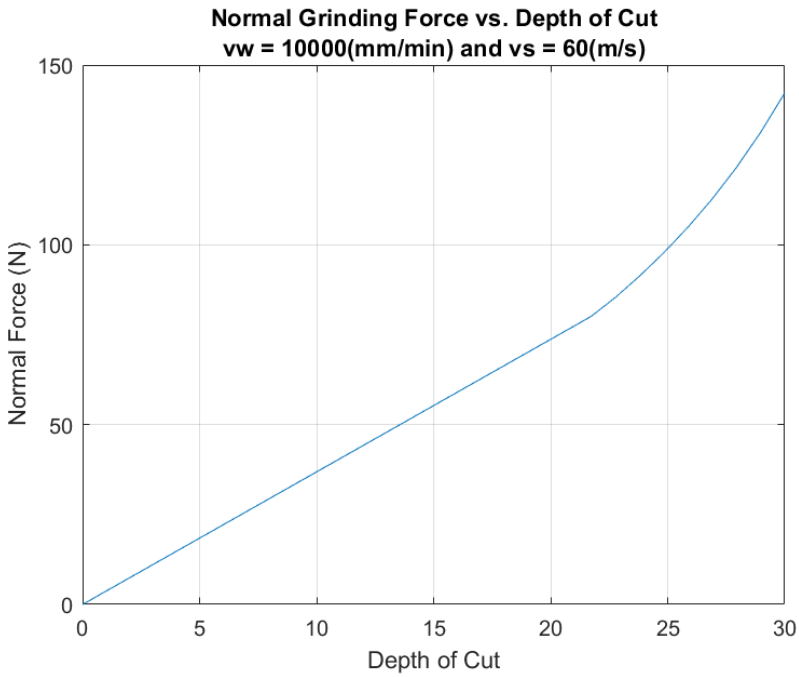

Figure 54 - Normal Force vs Depth of Cut - Vw = 10000mm/min, Vs = 60m/s

As can be seen in Figure 56 on the following page, the coefficient B vs. Residual stress has a linear relationship until the phase transformation begins to occur. An exponential growth can then be seen, as residual stress begins to plateau (slight linear increase as the hardness factor increases), while coefficient B increases in the same fashion as normal grinding force (coefficient B is a function of power and thus an inherited function of normal force).

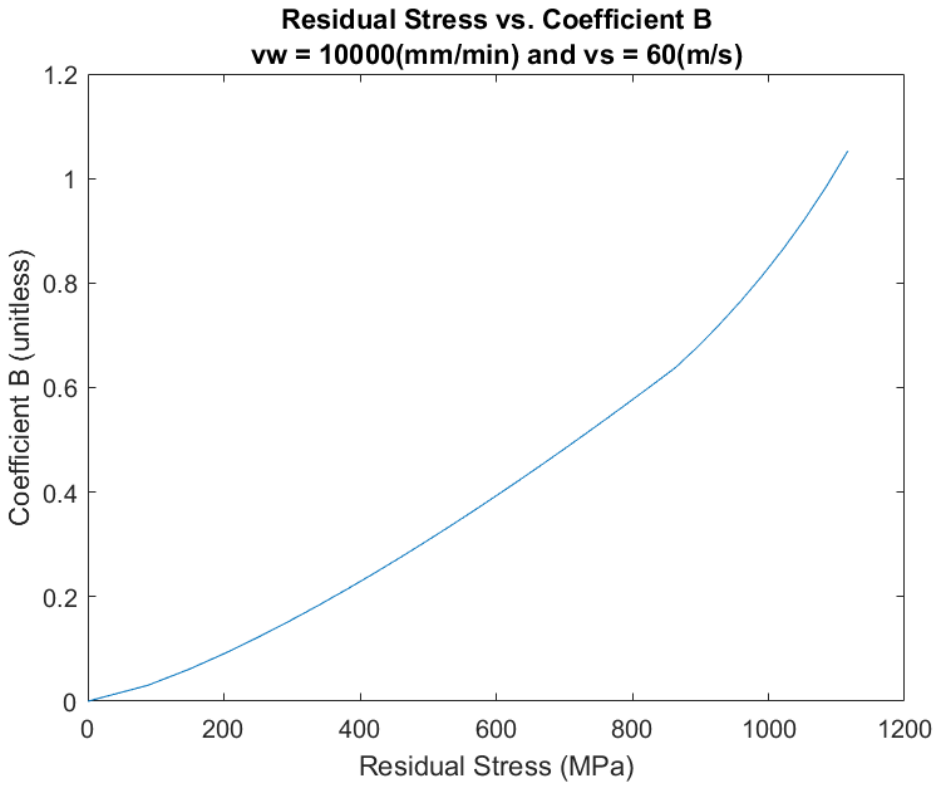

Figure 56 – Coefficient B vs. Residual Stress - Vw = 10000mm/min, Vs = 60m/s

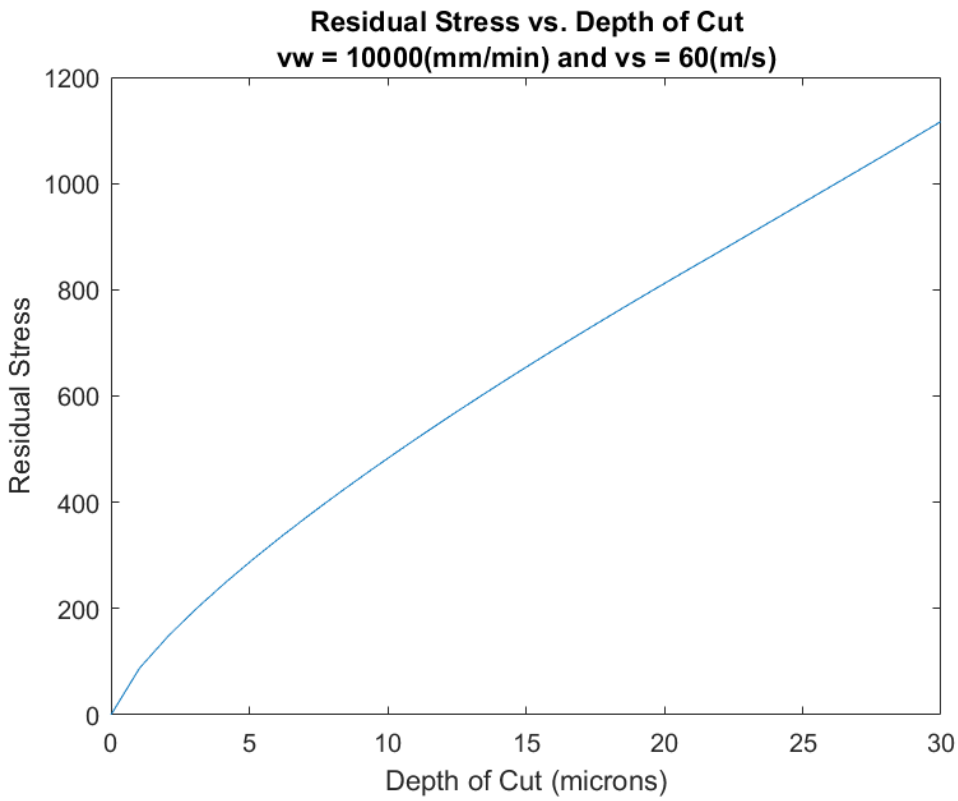

Figure 55 - Residual Stress vs. Depth of Cut - Vw = 10000mm/min, Vs = 60m/s

	Depth of Cut (microns)	Normal Force (N)	Tangen Force (N)	Temperature (°C)	RStress (MPa)	Coeff. B	VolWorkRemoved (mm^3)	SurfRough
1	0	0	0	25.0000	0	0	0	Inf
2	1.0345	3.8131	1.6899	78.7843	88.0445	0.0304	0.0464	0.1376
3	2.0690	7.6262	3.3798	115.4540	148.0727	0.0608	0.0928	0.0688
4	3.1034	11.4393	5.0697	147.6017	200.6982	0.0913	0.1392	0.0459
5	4.1379	15.2524	6.7596	177.1249	249.0276	0.1217	0.1856	0.0344
6	5.1724	19.0655	8.4495	204.8386	294.3947	0.1521	0.2320	0.0275
7	6.2069	22.8786	10.1395	231.1906	337.5327	0.1825	0.2784	0.0229
8	7.2414	26.6918	11.8294	256.4616	378.9012	0.2129	0.3248	0.0197
9	8.2759	30.5049	13.5193	280.8426	418.8128	0.2433	0.3712	0.0172
10	9.3103	34.3180	15.2092	304.4714	457.4929	0.2738	0.4176	0.0153
11	10.3448	38.1311	16.8991	327.4513	495.1109	0.3042	0.4640	0.0138
12	11.3793	41.9442	18.5890	349.8628	531.7983	0.3346	0.5104	0.0125
13	12.4138	45.7573	20.2789	371.7699	567.6601	0.3650	0.5568	0.0115
14	13.4483	49.5704	21.9688	393.2247	602.7816	0.3954	0.6032	0.0106
15	14.4828	53.3835	23.6587	414.2705	637.2334	0.4259	0.6496	0.0098
16	15.5172	57.1966	25.3486	434.9434	671.0748	0.4563	0.6960	0.0092
17	16.5517	61.0097	27.0385	455.2743	704.3564	0.4867	0.7424	0.0086
18	17.5862	64.8228	28.7285	475.2898	737.1216	0.5171	0.7888	0.0081
19	18.6207	68.6359	30.4184	495.0129	769.4083	0.5475	0.8352	0.0076
20	19.6552	72.4490	32.1083	514.4638	801.2493	0.5779	0.8816	0.0072
21	20.6897	76.2621	33.7982	533.6604	832.6739	0.6084	0.9280	0.0069
22	21.7241	80.0753	35.4881	552.6183	863.7080	0.6388	0.9744	0.0066
23	22.7586	85.3262	37.6094	571.3519	895.1841	0.6770	1.0374	0.0063
24	23.7931	91.2818	39.9420	589.8736	926.6856	0.7190	1.1084	0.0060
25	24.8276	97.7642	42.4327	608.1951	958.1206	0.7638	1.1854	0.0057
26	25.8621	104.8720	45.1110	626.3266	989.5506	0.8120	1.2695	0.0055
27	26.8966	112.7212	48.0118	644.2777	1.0210e+03	0.8642	1.3619	0.0053
28	27.9310	121.4483	51.1759	662.0569	1.0527e+03	0.9212	1.4641	0.0051
29	28.9655	131.2146	54.6517	679.6723	1.0845e+03	0.9837	1.5778	0.0049
30	30.0000	142.2099	58.4963	697.1310	1.1167e+03	1.0529	1.7051	0.0047

Table 2 - Results Summary - Vw = 10000mm/min, Vs = 60m/s

G = 40.7921 (Grinding Ratio)

Table 2 above shows the tabulated results from the model for the stipulated conditions. As most other variables are dependent on the force measurement, a similar case of normal force growth (as discussed two pages previously) can be seen for tangential force, coefficient B, and volume of work removed.

The same results template as specified above are now shown for a wheel speed of 45 m/s and a work speed of 12500 mm/min. The exponential growth of normal force begins at a lower depth of cut as the work speed is increased. A lower wheel speed results in a lower magnitude of force but does not affect the shape of the increasing curve.

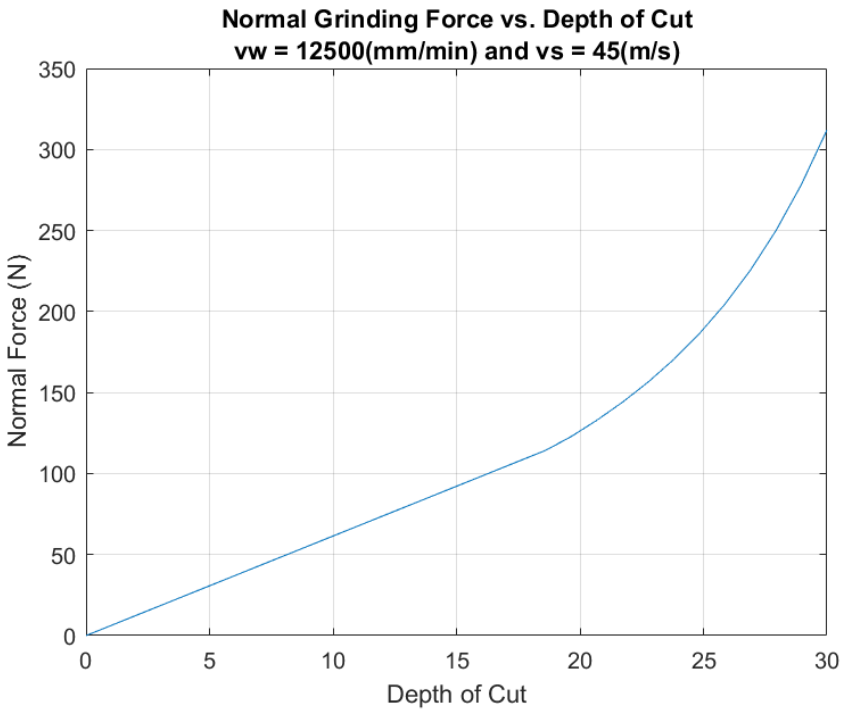

Figure 57 - Normal Force vs Depth of Cut - Vw = 12500mm/min, Vs = 45m/s

Residual stress vs coefficient B maintains a linear relationship up until phase transition occurs and increases exponentially from there, as earlier results may have suggested. The rate of residual stress increase becomes greater as the depth of cut is increased, and higher work speeds suggest a larger rate of increase as well. Grinding ratio (volume of workpiece removed to volume of wheel removed) decreased as the work-speed is increased, suggesting less wheel wear occurs at higher work-speeds.

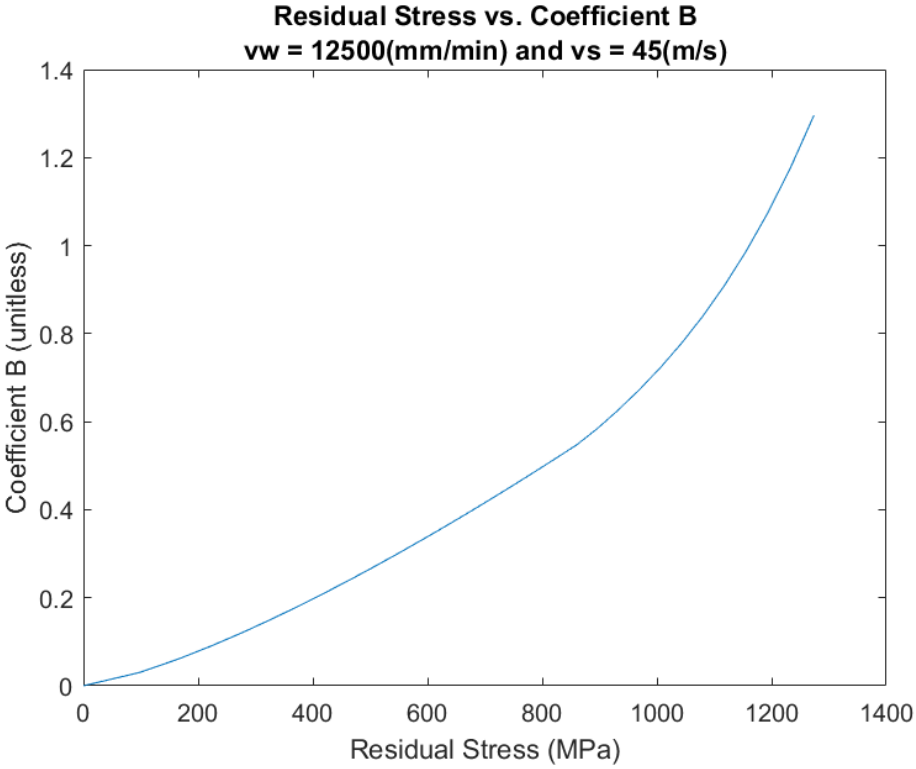

Figure 58 – Coefficient B vs. Residual Stress - Vw = 12500mm/min, Vs = 45m/s

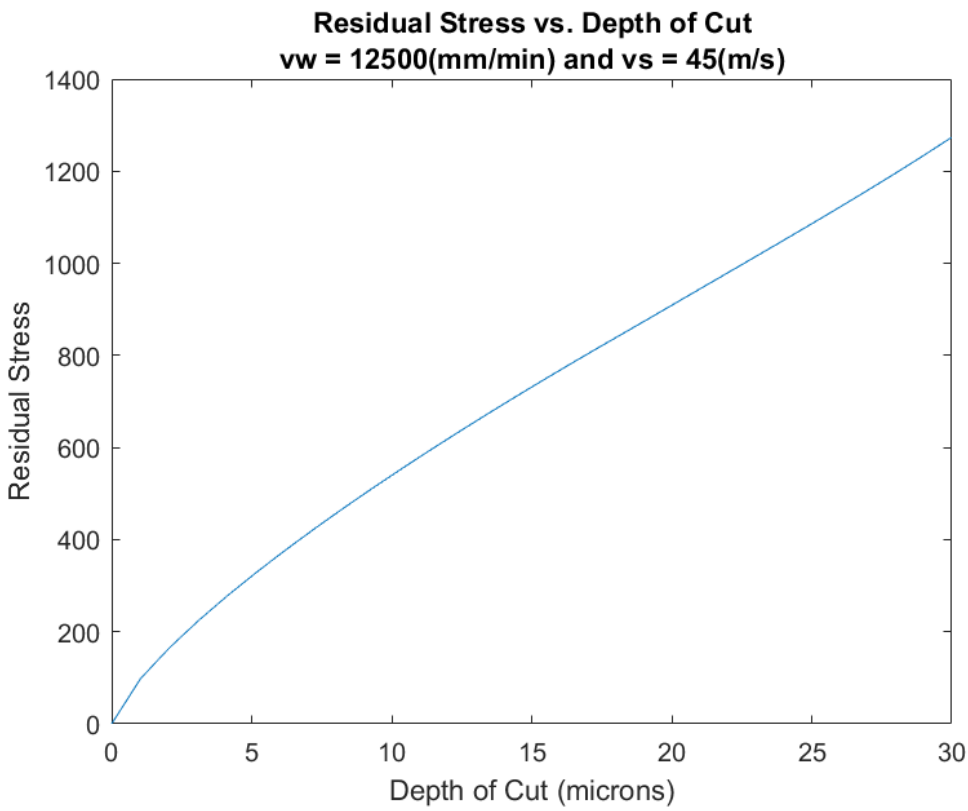

Figure 59 - Residual Stress vs. Depth of Cut - Vw = 12500mm/min, Vs = 45m/s

	Depth of Cut (microns)	Normal Force (N)	Tangen Force (N)	Temperature (°C)	RStress (MPa)	Coeff. B	VolWorkRemoved (mm^3)	SurfRough
1	0	0	0	25.0000	0	0	0	Inf
2	1.0345	6.3552	2.8165	85.1327	98.4368	0.0304	0.0773	0.2289
3	2.0690	12.7104	5.6330	126.1307	165.5503	0.0608	0.1547	0.1144
4	3.1034	19.0655	8.4495	162.0728	224.3874	0.0913	0.2320	0.0763
5	4.1379	25.4207	11.2661	195.0809	278.4213	0.1217	0.3093	0.0572
6	5.1724	31.7759	14.0826	226.0657	329.1433	0.1521	0.3867	0.0458
7	6.2069	38.1311	16.8991	255.5281	377.3731	0.1825	0.4640	0.0381
8	7.2414	44.4863	19.7156	283.7819	423.6244	0.2129	0.5413	0.0327
9	8.2759	50.8414	22.5321	311.0408	468.2470	0.2433	0.6187	0.0286
10	9.3103	57.1966	25.3486	337.4585	511.4926	0.2738	0.6960	0.0254
11	10.3448	63.5518	28.1652	363.1508	553.5508	0.3042	0.7733	0.0229
12	11.3793	69.9070	30.9817	388.2076	594.5686	0.3346	0.8507	0.0208
13	12.4138	76.2621	33.7982	412.7005	634.6633	0.3650	0.9280	0.0191
14	13.4483	82.6173	36.6147	436.6877	673.9303	0.3954	1.0053	0.0176
15	14.4828	88.9725	39.4312	460.2176	712.4485	0.4259	1.0827	0.0163
16	15.5172	95.3277	42.2477	483.3306	750.2844	0.4563	1.1600	0.0153
17	16.5517	101.6829	45.0642	506.0613	787.4944	0.4867	1.2373	0.0143
18	17.5862	108.0380	47.8808	528.4393	824.1270	0.5171	1.3147	0.0135
19	18.6207	114.3932	50.6973	550.4904	860.2246	0.5475	1.3920	0.0127
20	19.6552	122.9446	54.1727	572.2372	896.6843	0.5851	1.4946	0.0120
21	20.6897	132.9765	58.0922	593.6996	933.2258	0.6274	1.6142	0.0114
22	21.7241	144.0653	62.3288	614.8952	969.6947	0.6732	1.7458	0.0109
23	22.7586	156.4430	66.9521	635.8400	1.0062e+03	0.7231	1.8919	0.0104
24	23.7931	170.3901	72.0462	656.5479	1.0428e+03	0.7781	2.0556	0.0100
25	24.8276	186.2460	77.7129	677.0319	1.0797e+03	0.8393	2.2405	0.0095
26	25.8621	204.4213	84.0754	697.3036	1.1170e+03	0.9080	2.4509	0.0092
27	26.8966	225.4139	91.2832	717.3735	1.1548e+03	0.9859	2.6921	0.0088
28	27.9310	249.8274	99.5172	737.2513	1.1934e+03	1.0748	2.9700	0.0085
29	28.9655	278.3953	108.9975	756.9458	1.2330e+03	1.1772	3.2921	0.0082
30	30.0000	312.0109	119.9922	776.4653	1.2737e+03	1.2959	3.6668	0.0079

Table 3 - Results Summary - Vw = 12500, Vs = 45m/s

G = 21.1117 (Grinding Ratio)

The final displayed dataset is where work-speed is set at 15000mm/min and wheel-speed is set at 30m/s. The trends in results described earlier continue and are exaggerated more so in this set of results, with no new findings apparent in these results.

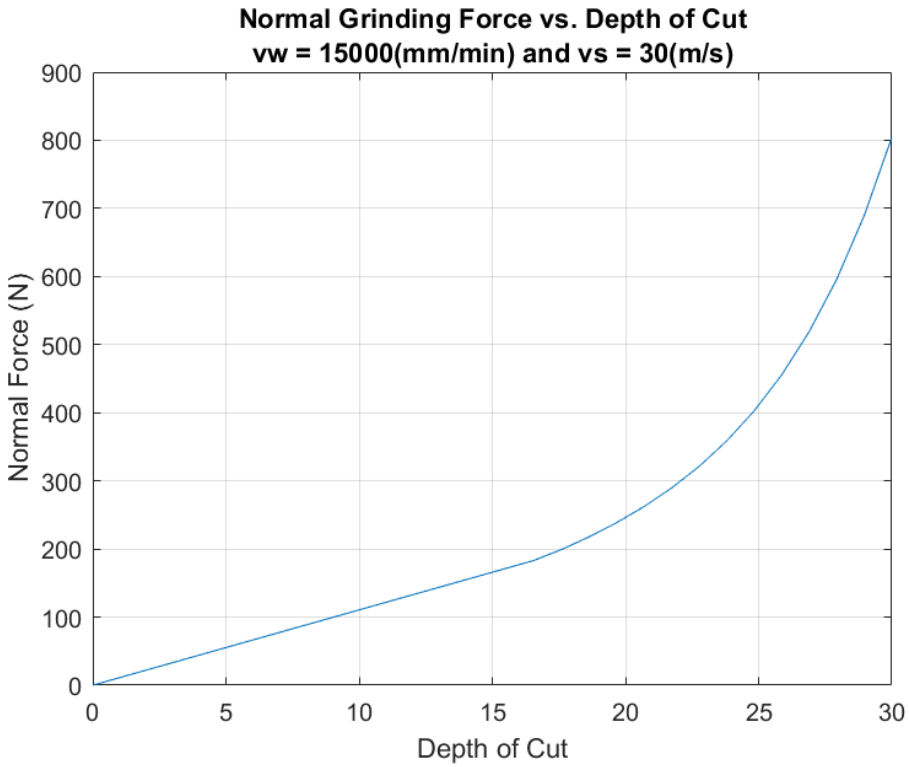

Figure 60 - Normal Force vs Depth of Cut - Vw = 15000mm/min, Vs = 30m/s

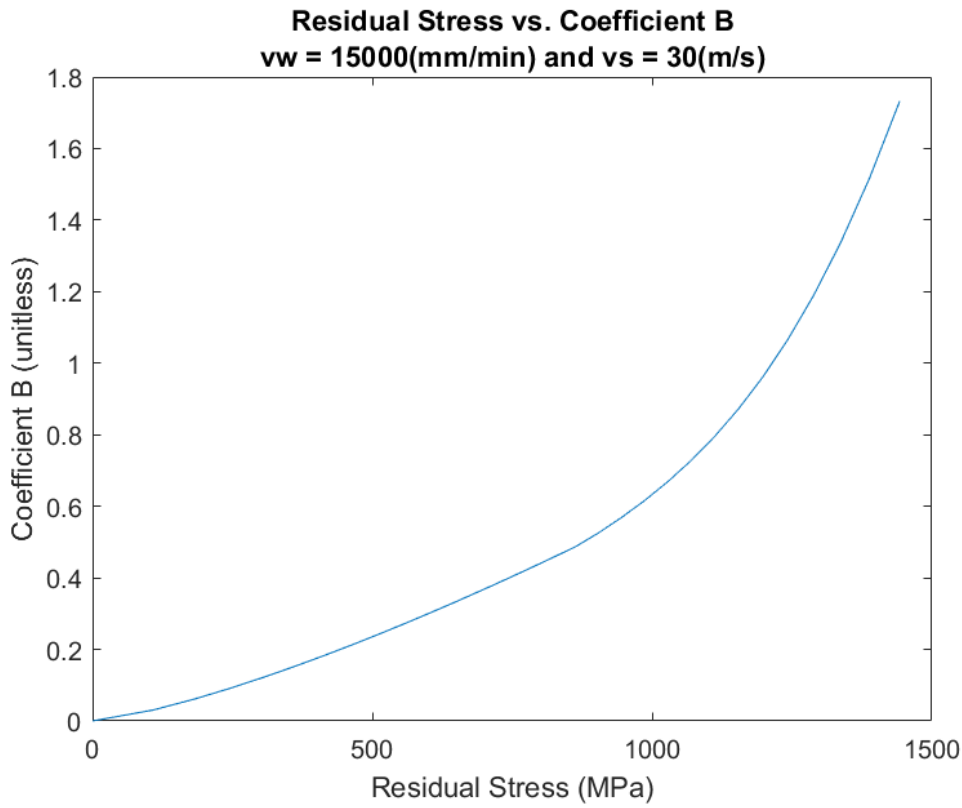

Figure 61 - Coefficient B vs Residual Stress, Vw = 15000mm/min, Vs = 30m/s

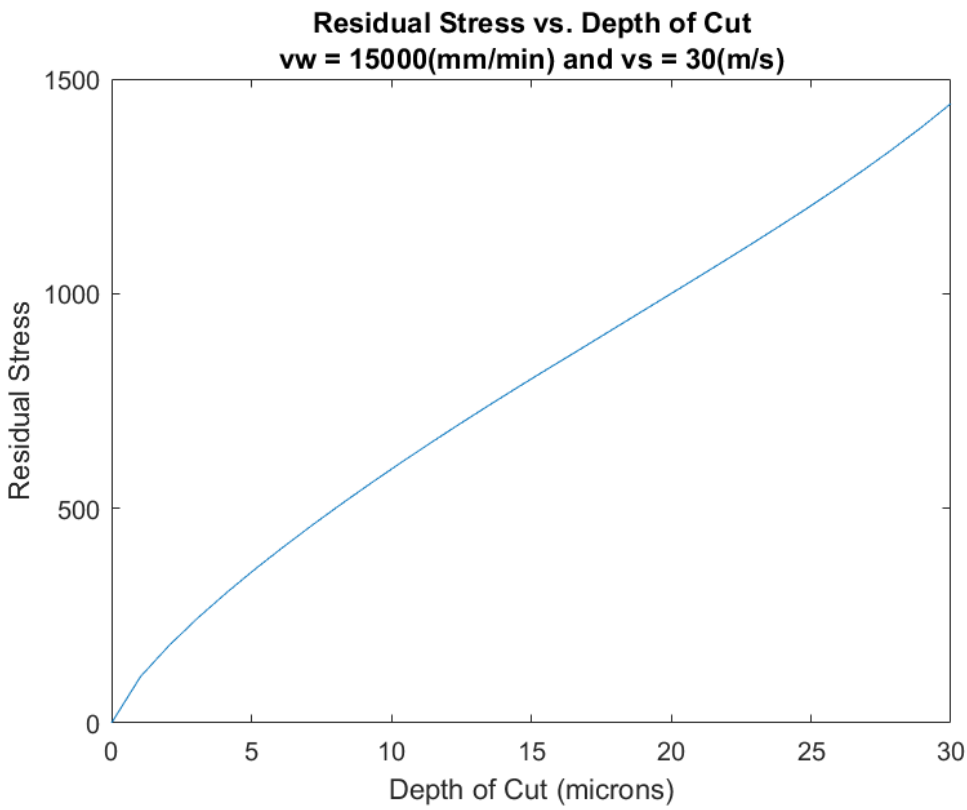

Figure 62 - Residual Stress vs. Depth of Cut - Vw = 15000mm/min, Vs = 30m/s

	Depth of Cut (microns)	Normal Force (N)	Tangen Force (N)	Temperature (°C)	RStress (MPa)	Coeff. B	VolWorkRemoved (mm^3)	SurfRough
1	0	0	0	25.0000	0	0	0	Inf
2	1.0345	11.4393	5.0697	90.8720	107.8321	0.0304	0.1392	0.4104
3	2.0690	22.8786	10.1395	135.7831	181.3513	0.0608	0.2784	0.2052
4	3.1034	34.3180	15.2092	175.1558	245.8041	0.0913	0.4176	0.1368
5	4.1379	45.7573	20.2789	211.3142	304.9953	0.1217	0.5568	0.1026
6	5.1724	57.1966	25.3486	245.2564	360.5584	0.1521	0.6960	0.0821
7	6.2069	68.6359	30.4184	277.5309	413.3915	0.1825	0.8352	0.0684
8	7.2414	80.0753	35.4881	308.4814	464.0573	0.2129	0.9744	0.0586
9	8.2759	91.5146	40.5578	338.3419	512.9388	0.2433	1.1136	0.0513
10	9.3103	102.9539	45.6275	367.2811	560.3121	0.2738	1.2528	0.0456
11	10.3448	114.3932	50.6973	395.4256	606.3845	0.3042	1.3920	0.0410
12	11.3793	125.8325	55.7670	422.8740	651.3173	0.3346	1.5312	0.0373
13	12.4138	137.2719	60.8367	449.7046	695.2389	0.3650	1.6704	0.0342
14	13.4483	148.7112	65.9065	475.9813	738.2537	0.3954	1.8096	0.0316
15	14.4828	160.1505	70.9762	501.7570	780.4483	0.4259	1.9488	0.0293
16	15.5172	171.5898	76.0459	527.0760	821.8954	0.4563	2.0880	0.0274
17	16.5517	183.0292	81.1156	551.9762	862.6569	0.4867	2.2272	0.0257
18	17.5862	199.0065	87.5468	576.4901	903.8980	0.5253	2.4187	0.0241
19	18.6207	217.5301	94.7418	600.6459	945.1318	0.5685	2.6392	0.0228
20	19.6552	238.3481	102.6251	624.4683	986.3130	0.6158	2.8857	0.0216
21	20.6897	262.0347	111.3690	647.9792	1.0276e+03	0.6682	3.1645	0.0205
22	21.7241	289.3018	121.1871	671.1979	1.0691e+03	0.7271	3.4832	0.0195
23	22.7586	321.0322	132.3441	694.1416	1.1111e+03	0.7941	3.8512	0.0187
24	23.7931	358.3222	145.1691	716.8261	1.1538e+03	0.8710	4.2799	0.0178
25	24.8276	402.5347	160.0707	739.2652	1.1974e+03	0.9604	4.7831	0.0171
26	25.8621	455.3682	177.5587	761.4717	1.2423e+03	1.0654	5.3776	0.0164
27	26.8966	518.9446	198.2696	783.4572	1.2888e+03	1.1896	6.0837	0.0158
28	27.9310	595.9240	223.0013	805.2322	1.3373e+03	1.3380	6.9259	0.0152
29	28.9655	689.6537	252.7582	826.8065	1.3884e+03	1.5165	7.9337	0.0147
30	30.0000	804.3645	288.8094	848.1890	1.4427e+03	1.7329	9.1428	0.0142

Table 4 - Results Summary - Vw = 15000mm/min, Vs = 30m/s

G = 9.3299 (Grinding Ratio)

5.4 Full Working Example of the System

Here, an example of the system working from initial start-up to final shut-down will be shown, primarily with screenshots and appropriate descriptions. This section links to the full integration section discussed earlier but differs in the sense that a specific example is shown step-by-step and results are analysed. Back-end work will be shown as needed too.

The user opens the MATLAB app and is met with the main menu as shown in Figure 63 below. The programmed buttons allow the user to either start the process or head to the guide to use page. The other buttons add minor functionality and ease of use that are aptly described by their names.

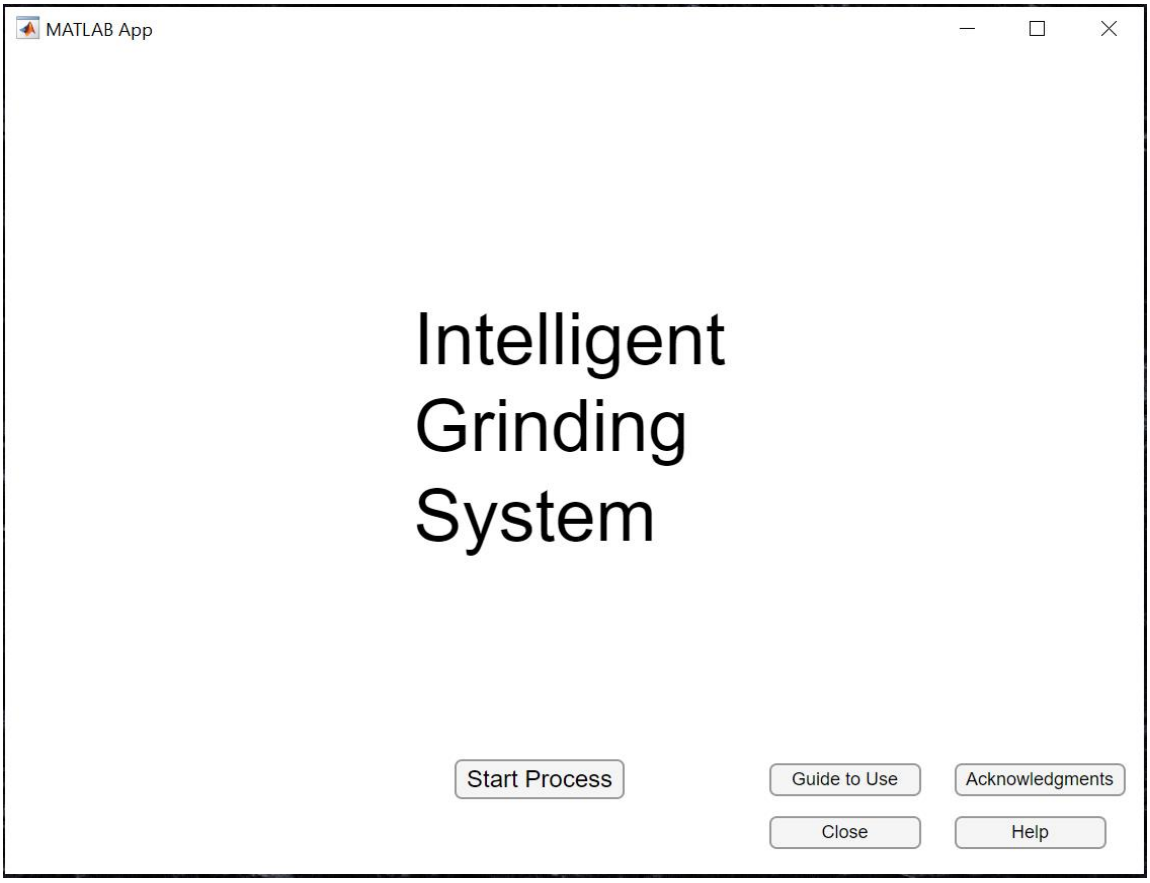

Figure 63 - IGS Main Menu (MATLAB)

If the operator is using the system for the first time the guide to use page is opened. Each step is read through one tab at a time and similar steps to what was described in the full integration of the system, are described on this page, as seen in Figure 64 below. The page can then be closed, and the user can then continue the system process by clicking on 'Start Process' on the main menu.

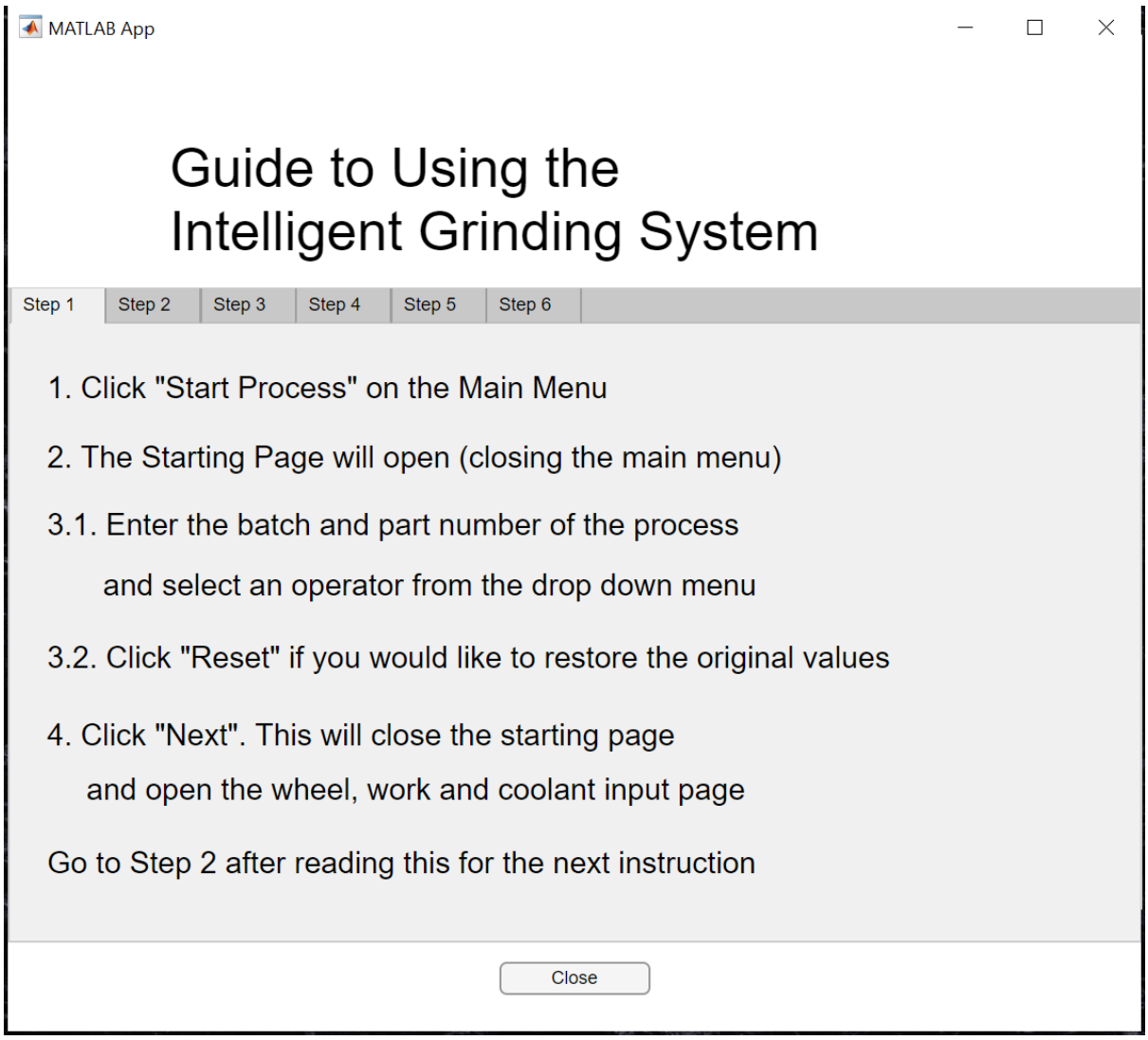

Figure 64 - Guide to Use Page (MATLAB)

The user is then met with the start page and must enter the batch and part number as well as a selection of operator from the drop-down menu (as seen in Figure 65 below). The inputs in this case are: 1, 3, 'Quintin'. These will be passed along the program (from applet to applet) and used for storage in the individual part record table of the database.

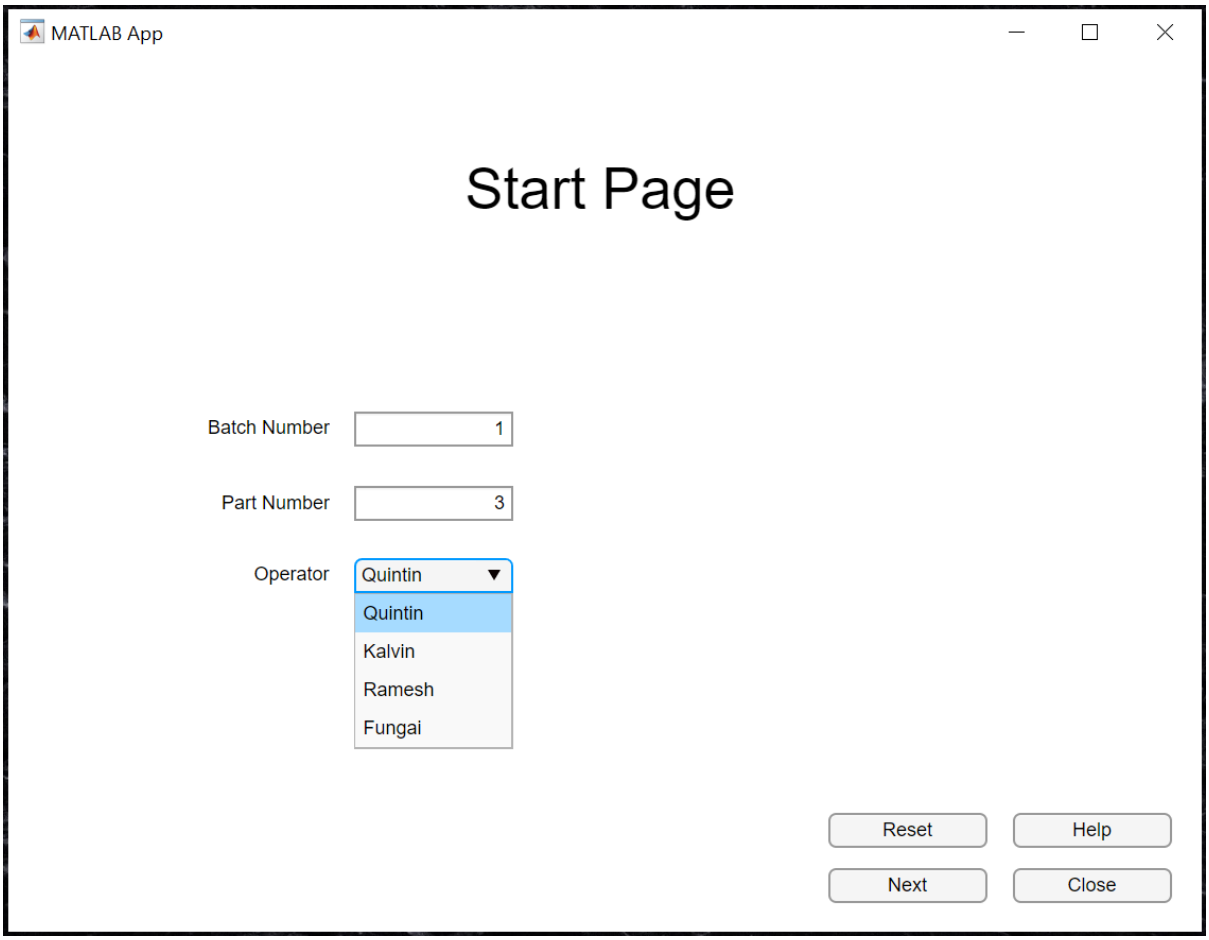

Figure 65 - IGS Start Page (MATLAB)

The next button is then pressed, and the user is taken to the material, wheel and workpiece coolant page, as seen in Figure 66 below. A wheel, material and coolant must be selected from the dropdown menu (if no coolant is in use, 0m/s can be entered for the coolant flow rate). The length, width and initial phase transformation percentage are then entered by the user (120mm, 200mm and 10% in this case). The flow rate can then be entered too (10m/s in this case). The selected wheel, material and coolant are 'BZ100R100BQ1/8', 'Ti-6Al-4V' and 'Water + 5%CuO' respectively. All the entries in the drop-down menus are acquired from the database through linkage code and automatically updated in the MATLAB GUI each time a new database entry is added, and subsequently, a new system run is initiated.

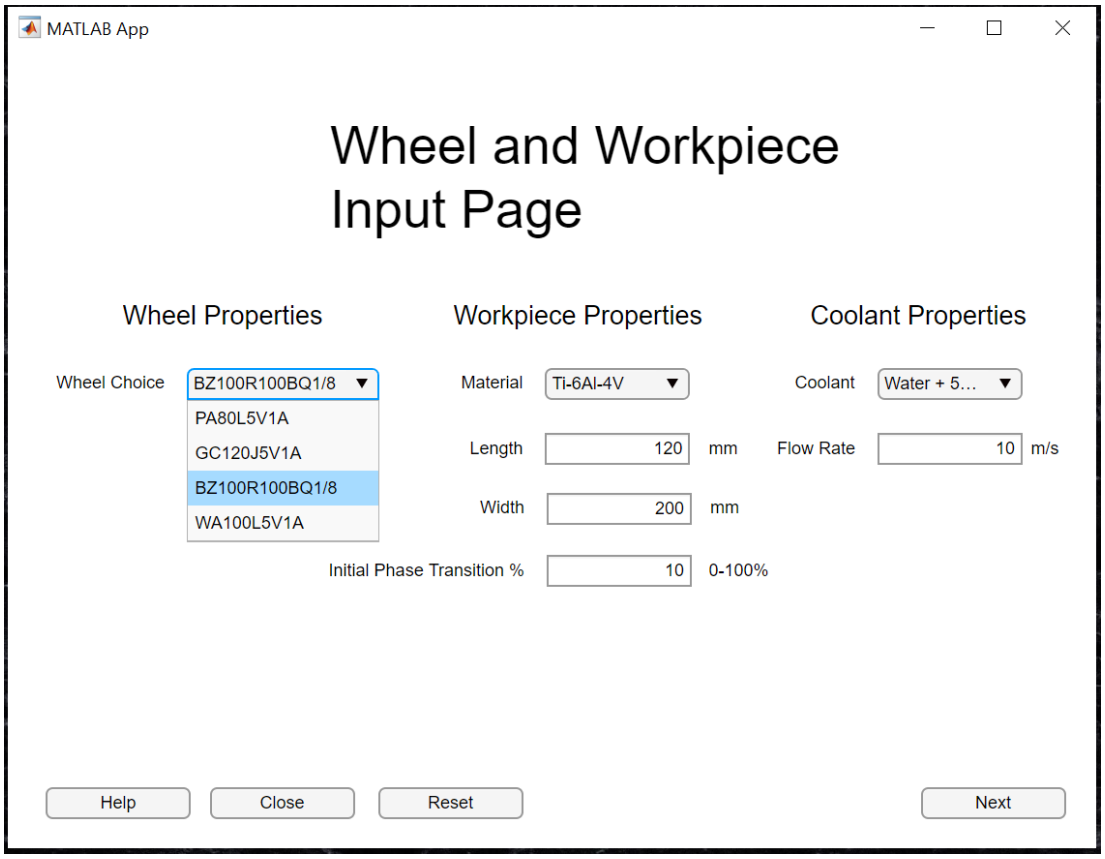

Figure 66 - IGS Wheel, Work, Coolant Input Page (MATLAB)

At this point, if the material or wheel or coolant that is desired is not in the database, the user will stop the process and direct themselves to the database GUI and input a wheel and/or a material and/or a coolant. The tables in the database linked to the system GUI can be seen in Figure 67 on the following page.

MaterialData

ID	MaterialName
1	Ti-6Al-4V
2	AISI E9310 Steel
3	Aluminium 6061
(New)	

WheelData

ID	Specification
1	BZ100R100BQ1/8
2	WA100L5V1A
3	GC120J5V1A
4	PA80L5V1A
(New)	

CoolantData

ID	CoolantName
5	Water
6	Water + 5%CuO
(New)	

Figure 67 - Material, Wheel, Coolant Tables (Access Database)

The user will then re-start the system and head back to wheel, workpiece and coolant page and re-enter and select the desired values. These desired values, along with the start page values, are passed along the applets too. The user then clicks next and is met with the grinding parameters and properties input page, as seen in Figure 68 below.

Various entries are required on this page and require a slightly deeper understanding of the grinding process. The typical values button takes the user to a page similar to the guide to use page, which shows values of typical grinding processes. The work speed, wheel speed, depth of cut, desired surface roughness, optimization selection, coefficient of friction, specific grinding energy and grinding angle entered in this example are 12500 mm/min, 30 m/s, 9 micrometres, 0 micrometres, 'None', 9.5 J/mm^3 and 30 degrees, respectively. Again, the other buttons offer simple functionality that has been coded in and is self-explanatory.

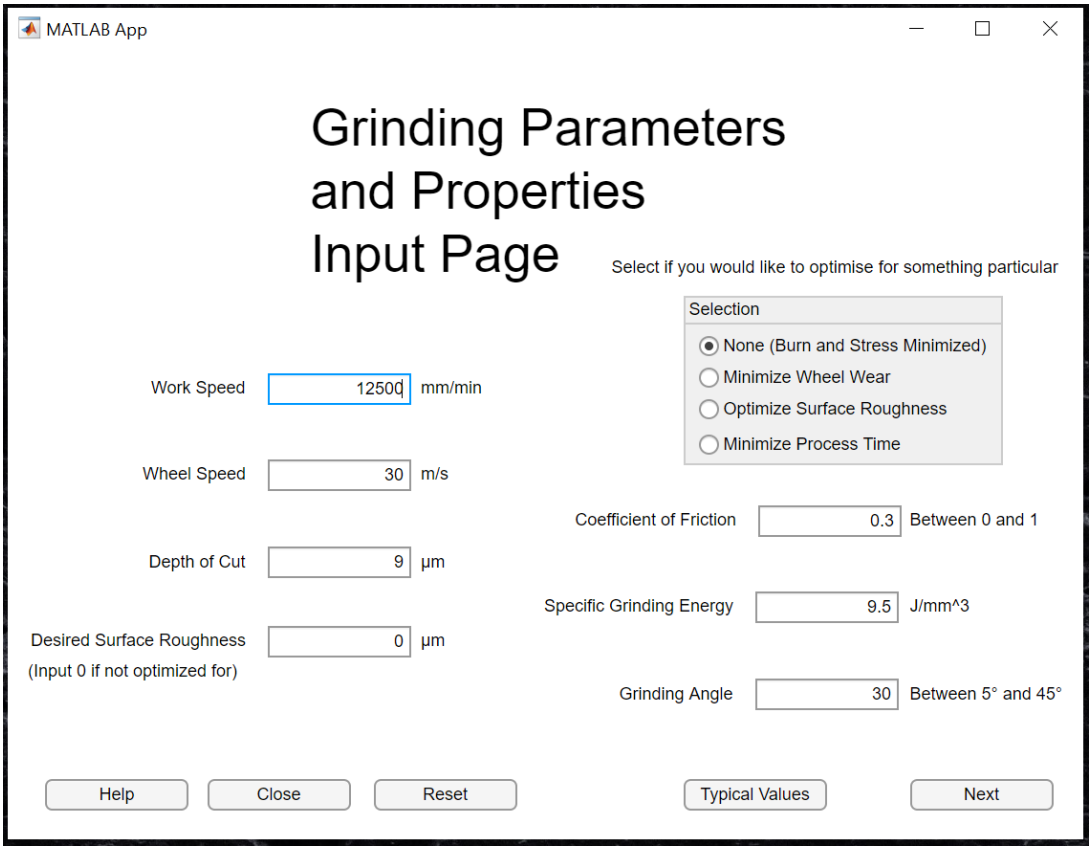

Figure 68 - IGS Grinding Parameters Input (MATLAB)

The next button is then pressed by the operator and they are taken to the output, recommendations and pre-grinding page. All previous input data from each user input is passed here. Firstly though, the active cycle table in the database is updated (Figure 69 below) with the data input above, then the selected material, wheel and coolant properties are acquired (selected) from their respective tables and brought to the program, where they are then used in the previously developed model. After the model and associated code has run (on start-up of this page opening), the outputs, calculated values and plots are updated. The recommendations are filled with what the user has input (as previously described, no feature correlation engine has been built in). Figure 70 shows the predicted temperature, residual stress, final phase transformation, coefficient B, surface roughness, wheel volume removal in one pass and grinding ratio for the user inputs stated. The calculated values shown in Figure 71 (following page) are of less importance but show some intermediate results that may be of interest to some operators.

ActiveCycleData

ID	WorkMaterial	WorkLength	WorkWidth	InitialTransPercent	Wheel	Coolant	Flowrate	Workspeed	Wheelspeed	DepthofCut
1	Ti-6Al-4V	120	200	0,1	BZ100R100BQ1	Water + 5%CuC	10	12500	30	9
* 1		0	0	0			0	0	0	0

Figure 69 - Updated Active Cycle (Access Database)

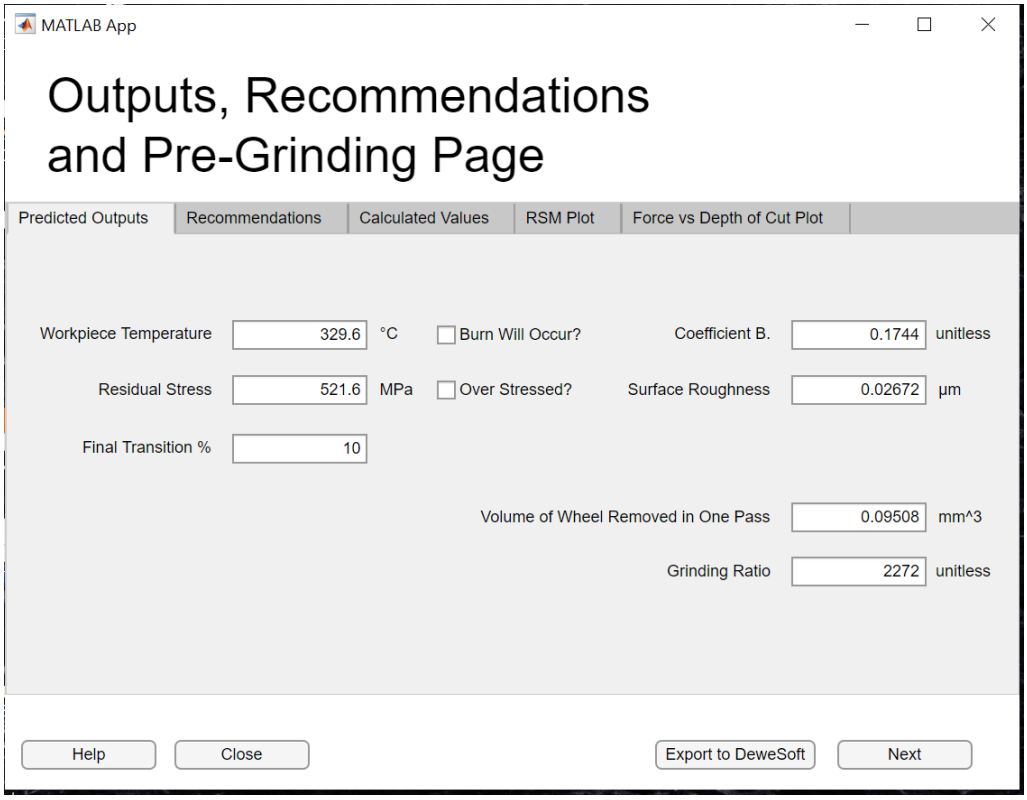

Figure 70 - IGS Outputs and Recommendations, Predicted Outputs (MATLAB)

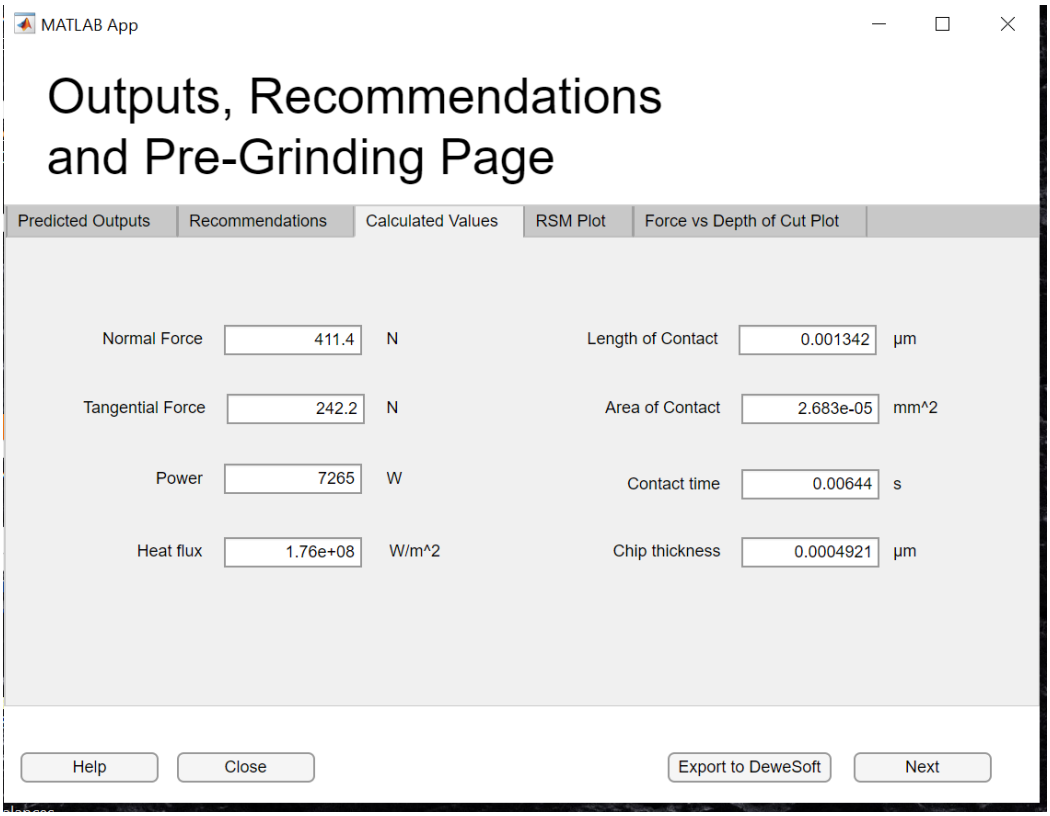

Figure 71 - IGS Outputs and Recommendations, Calculated Values (MATLAB)

After the user is done, they must click 'Export to DeweSoft', where relevant data (only the maximum coefficient B, temperature, residual stress and predictions of these) are exported to a text file. This text file is read as a channel by DeweSoft, which imports the data and displays it to the user on the user interface panel. After clicking Export to DeweSoft, the user must minimize the output page and open the DeweSoft setup file included in the package. This setup file includes a simple user interface allowing the user to start, stop and export the data. In the case include here, the DeweSoft file is that of a data file rather than a set-up file. This is because a live run cannot be done without actual equipment and the data file can show the functioning of the DeweSoft program in the same fashion as that of a normal system run, just without the need for all the hardware.

Figures 73 and 72 show the home-page, and run-page of the DeweSoft data-file (the set-up file has been created in a very similar fashion). The user must only run the process and if the temperature or residual stress exceeds the allowable values, they must stop the process. Finally the data must simply be exported to the location specified in the software package, 'CurrentGrindingData'. DeweSoft can then be exited and the MATLAB app can be returned to.

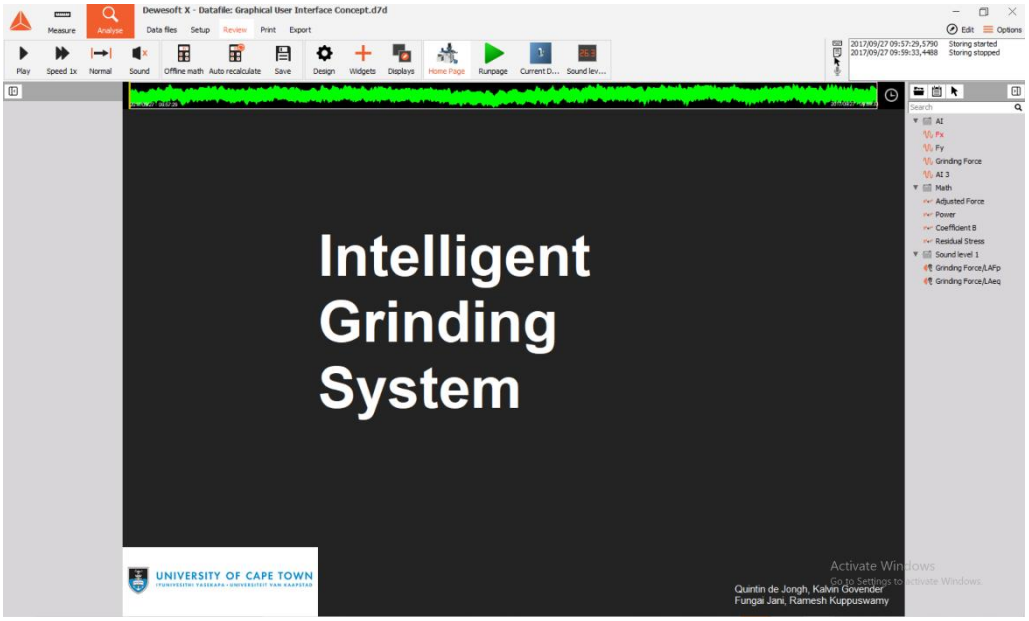

Figure 73 - IGS Homepage (DeweSoft)

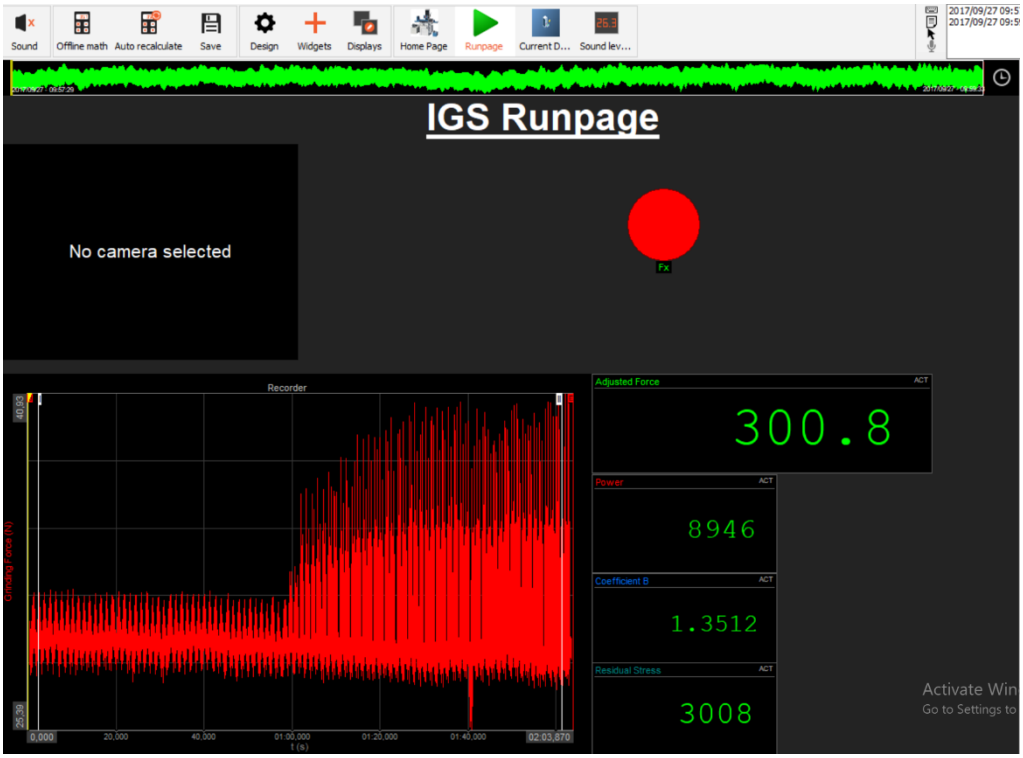

Figure 72 - IGS Run-page (DeweSoft)

The user must then click next on the output page. This will pass all previous information (inputs and outputs) to the analysis of data page. Before the analysis page is opened, the reference cycle data and individual part record tables are updated in the database, as seen in Figures 74 and 75 below.

ReferenceCycleData

ID	WorkMaterial	WorkLength	WorkWidth	InitialTransPercent	Wheel	Coolant	Flowrate	Workspeed	Wheelspeed	DepthofCut	Co
1	Ti-6Al-4V	100	200	0,1	BZ100R100BQ1	Water	10	11000	30	9	
2	Ti-6Al-4V	100	20	0,1	BZ100R100BQ1	Water	10	13000	20	9	
3	Ti-6Al-4V	100	200	0,1	BZ100R100BQ1	Water	10	13000	20	6	
4	Ti-6Al-4V	100	10	0,1	BZ100R100BQ1	Water	10	12500	30	9	
5	Ti-6Al-4V	100	20	0,1	BZ100R100BQ1	Water	0	12500	30	6	
6	Ti-6Al-4V	200	300	0,1	BZ100R100BQ1	Water	5	12500	30	9	
7	Ti-6Al-4V	100	200	0,1	BZ100R100BQ1	Water	10	13500	30	9	
8	Ti-6Al-4V	100	260	0,1	BZ100R100BQ1	Water	0	12500	30	9	
9	Ti-6Al-4V	100	200	0,1	BZ100R100BQ1	Water	50	12500	30	9	
10	AISI E9310 Steel	100	30	0,2	PA80L5V1A	Water + 5%CuC	10	12000	30	14	
11	Ti-6Al-4V	120	200	0,1	BZ100R100BQ1	Water + 5%CuC	10	12500	30	9	
(New)		0	0	0			0	0	0	0	

Figure 74 - Updated Reference Cycle Table (Access Database)

IndividualPartRecord

ID	PartNo	BatchNo	Operator	ReferenceCycleNumber	MaxFn	MaxFt	MaxCoeffB	MaxResidStr	MaxTemp	GrindingRatio
2	0	0	Quintin	2	64,1742	37,7773	0	531153000	335,647	1456,2
3	1	1	Fungai	3	434,583	273,208	0,126096	407,979	257,81	1433,57
4	1	1	Quintin	4	20,5687	12,1081	0,174357	521,607	329,614	2271,68
5	2	2	Fungai	5	28,1466	16,569	0,119297	404,568	255,655	2213,43
6	3	3	Ramesh	6	605,66	356,533	0,171136	512,702	323,987	2314,44
7	5	5	Kalvin	7	442,225	304,546	0,203031	538,197	340,098	2113,19
8	0	0	Quintin	8	543,229	319,781	0,17711	529,218	334,424	2236,37
9	3	3	Quintin	9	411,373	242,162	0,174357	521,607	329,614	2271,68
10	42	42	Kalvin	10	82,7289	51,4187	0,257093	523,47	152,321	3008,12
11	1	1	Quintin	11	411,373	242,162	0,174357	521,607	329,614	2271,68
(New)	0	0		0	0	0	0	0	0	0

Figure 75 - Updated IPR (Access Database)

Again, all relevant material, wheel and coolant properties are selected from the database in order to run portions of the model required in the analysis page. After this, the data exported from DeweSoft, is imported into MATLAB and the noise is eliminated using the methods described in earlier sections. The user can then interact with the graph using the zoom, pan and export buttons as well as choosing to input a time range which outputs the kurtosis coefficient and the associated wheel wear for the portion selected by the user. It is recommended that the user does the kurtosis analysis over one pass only. The coefficient B vs residual stress plot and temperature vs. time plot are shown on the subsequent tabs along with the passed-on values of maximum coefficient B, residual stress and temperature. Once the interaction with the analysis page is complete, the user can then choose to save the cycle as optimal, which will add to the optimised cycle table in the database (as seen in Figure 77). Once the user clicks 'Close' the system run is complete, and all necessary tasks have been completed.

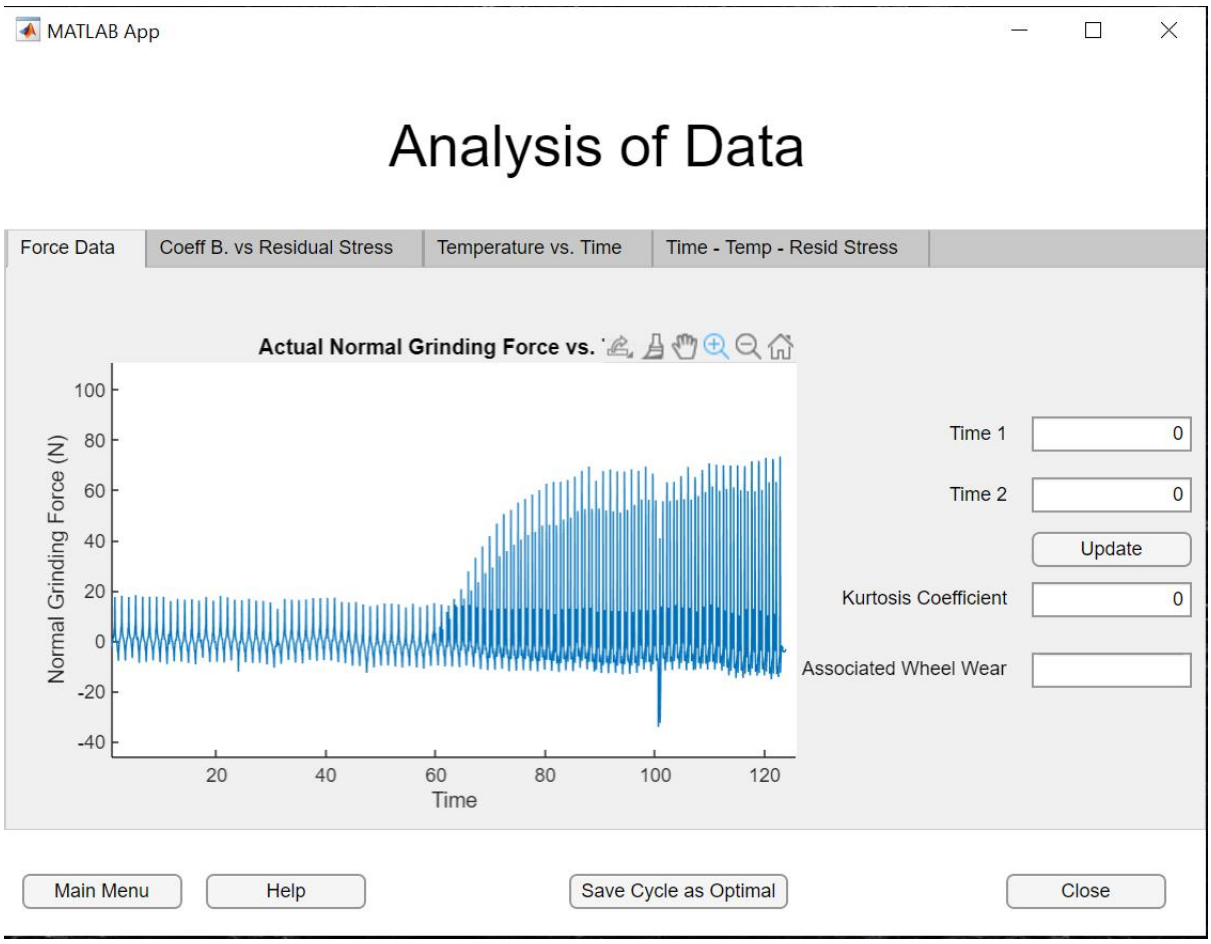

Figure 76 - IGS Analysis Page (MATLAB)

OptimisedCycleData

ID	WorkMaterial	WorkLength	WorkWidth	InitialTransPercent	Wheel	Coolant	Flowrate	Workspeed	Wheelspeed	DepthofCut	C
1	Ti-6Al-4V	100	200	0,1	BZ100R100BQ1	Water	10	13000	20	6	
2	Ti-6Al-4V	100	200	0,1	BZ100R100BQ1	Water	50	12500	30	9	
3	Ti-6Al-4V	120	200	0,1	BZ100R100BQ1	Water + 5%CuC	10	12500	30	9	
(New)		0	0	0			0	0	0	0	

Figure 77 - Updated Optimised Table (Access Database)

6 Conclusions

The development of an intelligent grinding system and its attributed components have been described throughout the report and many conclusions relating to the findings of research and development of system, can be produced. The major outcomes of the project and its significance will be highlighted below, along with statements of aim achievements. The conclusions will be divided into sections relating to the chapters in the report.

Relating to the development of the model:

- The model produces the required outputs of workpiece temperature, coefficient B and residual stress formation, based on the inputs of workpiece material (and inherited properties), wheel, and grinding parameters.
- The model produces results over a wide variety of conditions.
- The model accounts for temperature dependant properties of workpieces and these effects define how grinding forces vary according to ease of grinding, either exhibiting a linear or exponential growth with depth of cut increasing.

The aim of developing an adaptable and reliable surface grinding model based on workpiece and wheel properties, along with grinding parameters, has thus been achieved.

Regarding the database:

- A database has been developed in Microsoft Access for the system and contains multiple tables that relate to storage of properties (and corresponding retrieval by the system) or storage of system inputs (cycle data tables) or system outputs (individual part records)
- The wheel, materials and coolant tables store all relevant properties (and temperature dependant properties) that are required by the developed model.
- The active, reference and optimised cycle data store inputs from the user to the system, with each relating to the current cycle, all runs of the system and the system runs that do not incur burn or overstressing, respectively.
- The individual part record stores all outputs from the system, reference cycle identification number and the operator name, part number and batch number.

This concludes that the aim of developing a structured database, relating to the developed model, that allows for effective storage, collection and retrieval of data, has been achieved.

Regarding the human-machine interfaces:

- GUIs have been created for the database, MATLAB app and DeweSoft run time program, and have been kept as minimal as possible in terms of user input, in order to facilitate ease of use and learning.
- The database GUI was created with Access Forms and allows for efficient viewing, editing, addition and deletion of elements in the database tables.
- The DeweSoft GUI allows for easy running of the grinding process and is used mainly for exporting data. Channels are set up to give the most important information to the user (coefficient B and maximum temperature). Live predication of coefficient B and maximum temperature are provided as per the initial empirical model.
- The most important GUI, the MATLAB GUI, contains the core of the system and links the user to the system, database and live DeweSoft program. The sequential order of the system guides the user through a series of steps to the output of the system, where important results are shown, before guiding the user to the DeweSoft app and then back to an analysis page where users can analyse the data.

Developing a human-machine interface that requires minimal learning time and allows for ease of use has been developed, and thus this aim of this has been completed.

For creating grinding 'mock' experimental data:

- The model was used to create a large amount of 'mock' experimental data for the Ti-6Al-4V alloy over a range of wheel speeds (30 – 60 m/s), work speeds (10000 – 15000 mm/min) and depths of cut (0 – 60 micrometres). This produces a large quantity of results for many features, namely: normal and tangential forces, surface temperatures, coefficient B, residual stress and surface roughness.

The creation of 'mock' experimental data to allow for machine learning has been achieved and was another aim of this project.

Regarding signal processing and Ti-6Al-4V:

- A discrete wavelet transformation and a moving average filter have been used to eliminate noise from acquired signals.

The processing (namely noise elimination) of raw signals has been completed and so this aim was met, alongside the aim of investigating and researching the Ti-6Al-4V alloy.

In reference to real-time analysis:

- DeweSoft human interaction has been facilitated with the importing of maximum allowable coefficient B and residual stress. This allows operators to manually stop the process before limits have been reached. Adjustments must then be made manually.
- Live grinding condition adjustments in DeweSoft have not been made. The reference section that follows this describes what could be done to allow for this to be completed.

Real time analysis was completed in the sense that limits were produced for the operator to manually adjust conditions, however the aim of live conditional monitoring and adjustment was not achieved due to the limited availability of time and access to grinding equipment (all work was theoretical in nature).

Overall, the project was a success in its course, as all viable aims were achieved, and results have been produced. The researched information and developed system can provide a core for future developments of machine intelligence in grinding, particularly for other students at the institution (should this project be taken further).

7 Recommendations

Following the completion of this project and report, and based on the methodology, results and conclusions, recommendations to future reviewers, researchers or developers can be made. Improvements have been identified that could make the system more optimal but that have not been implemented due to scope, time or knowledge constraints. These have been included in this section as well.

Recommendations based on developments and results:

- Temperature dependant material properties must be taken into account when considering workpiece material behaviour in the grinding process.
- Work-speeds and wheel-speeds are important inputs that are easily adaptable for optimization and should be used as such.
- Feature selection in an intelligent system should be limited to the most descriptive yet simplest to acquire e.g. surface temperature relates to burn and coefficient B relates to residual stress formation.
- In order for an intelligent system to be effective, an accurate model must be developed that can describe the process over a wide variety of grinding conditions.
- After acquiring live signals from a machining process, they must be processed in order to correctly analyse them, with noise the most aspect to process/eliminate.
- When creating a data storage structure (database), a structure that directly links to the developed model, must be created.
- The human-machine interface must be user-friendly, simple to use and quick to learn (along with its corresponding interface)

Improvements and optimisations that can be made:

- To provide an accurate simulation of the grinding process, the grain-force micro-analytical model should be implemented (although this will be a large undertaking by itself).
- Experiments should be run to confirm the viability of the models over various conditions.
- A statistical model such as the Werner model should be implemented in tandem with the developed model and it can be used to confirm results and create a more efficient model when grinding a single material over various conditions. It can also be used as a data source to teach a neural network.
- The coolant properties and effects should be included and accounted for in the developed model. It can then be connected to the database where it is already accounted for.

- Dress data and prediction should be included in the optimization engine as well.
- A feature correlation engine (most importantly machine learning) should be built and implemented into the system in order for the system to make recommendations and accurate optimisation/prediction that learns from iterations of system runs.
- The DeweSoft GUI could be eliminated and live signals could be read in through MATLAB's data acquisition toolbox so that the system is run on a single interface (making for easier implementation of algorithms and grinding control). This could be done the opposite way around where all MATLAB code is programmed into the DeweSoft (although this may prove more difficult).
- The response surface methodology could be implemented more so into the system to allow for greater optimization of conditions (this would not be a replacement for feature correlation but rather an extra functionality of the system).

8 Reference List

[1] A.V. de Mello, R.B de Silva, A.R. Machado, R.V Gelamo, A.E. Diniz and R.F.M de Oliveira, "Surface Grinding of Ti-6Al-4V Alloy with SiC Abrasive Wheel at Various Cutting Conditions", *Procedia Manufacturing,* vol 10, pp. 590 – 600, 2017.

[2] H. Jamshidi and E. Budak, "An analytical grinding force model based on individual grit interaction", Journal of Materials Processing Tech., no 283, pp. 1 – 15, 2020.

[3] W.B. Rowe, "Principles of Modern Grinding Technology", 1st edition, Oxford, Elsevier, 2009, pp.2-3

[4] M.N. Morgan, R. Cai, A. Guidottti, D.R. Allanson, J.L. Moruzzi and W.B. Rowe, "Design and implementation of an intelligent grinding assistant system", *International Journal of Abrasive Technology,* vol. 1 no. 1, pp. 106 - 134, 2007.

[5] G. Amitay, S. Malkin and Y. Koren, "Adaptive Control Optimization of Grinding", *Journal of Engineering for Industry,* vol 103 no. 1, pp. 131 – 136, Feb. 1981.

[6] S.S. de Rocha, G.L. Adabho, G.E.P. Henriques and M.A.dA. Nobilo, "Vickers Hardness of Cast Commercially Pure Titanium and Ti-6Al-4V Alloy Submitted to Heat Treatments", Brazilian Dental Journal, vol. 17 no. 2, pp. 126-129, 2006.

[7] R. Pederson, "Microstructure and Phase Transformation of Ti-6Al-4V", Licentiate Thesis, Dept. of Applied Physics and Mechanical Engineering, Lulea University of Technology, Lulea, Sweden, 2002.

[8] T. Choi, Y.C. Shin, "Generalized Intelligent Grinding Advisory System", *International Journal of Production Research,* vol 45 no. 8, pp. 1899-1932, 9 March 2007.

[9] Q.O. de Jongh, "Interim Report 2020, Development of an Intelligent Grinding System", Department of Mechanical Engineering, University of Cape Town, South Africa, June 2020.

[10] P. Lezanski, "A data-driven predictive model of the grinding wheel wear using the neural network approach", *Journal of Mechanical Engineering,* vol 17 no 4, pp. 69 - 82, 2017.

[11] Y. Li, "Intelligent Selection of Grinding Conditions", Doctor of Philosophy, Liverpool John Moores University, Liverpool, England, September 1996.

[12] B.S. Linke, I. Garretson, F. Torner, "Grinding energy modelling based on friction, plowing and shearing", Journal of Manufacturing Science and Engineering, vol 139 no 12, pp. 1 – 11, July 2017.

[13] A.O. Alao, “PhD Research Proposal, A real time-based predictive modelling of grinding induced residual stress for aerospace components”, Department of Mechanical Engineering, *University of Cape Town, Cape Town, South Africa,* July 2019.

[14] G. Werner, “Influence of work material on grinding forces”, Annals of CIRP, vol 27, pp. 243 -248, 1978.

[15] V.K. Mishra and K. Salonitis, “Empirical estimation of grinding specific forces and energy based on a modified Werner grinding model”, 14th CIRP Conference on Modelling of Machining Operations, Procedia CIRP 8, pp. 287 – 292, 2013.

[16] A.S Patil, V.S. Ingle, S.Y. More and S.M. Nathe, “Machining Challenges in Ti-6Al-4V.-A Review”, International Journal of Innovations in Engineering and Technology, vol. 5 no 4, August 2015.

[17] L. Hobbs, S. Hillson, S. Lawande and P. Smith, ‘”16 – Oracle Data Mining”, Oracle 10g Data Warehousing, pp. 725-755, 2005.

[18] E.A.A. Maksoud, S. Barakat, M. Elmogy, “Chapter 9 – Medical Images Analysis Based on Multilabel Classification”, Machine Learning in Bio-Signal Analysis and Diagnostic Imaging, pp.209-245, 2019.

[19] T.W. Liao, “Feature extraction and selection from acoustic emission signals with an application in grinding wheel condition monitoring”, Engineering Applications of Artificial Intelligence, vol. 23 no.1 , pp. 74-84, 28 October 2009.

[20] P. Lezanski, “A data-driven predictive model of the grinding wheel wear using the neural network approach”, *Journal of Mechanical Engineering,* vol 17 no 4, pp. 69 - 82, 2017.

[21] R. Kuppuswamy and K.A. Airey, “Feature extraction on an intelligent polycrystalline diamond insert clock testing method and prediction of product performance”, *Journal of Process Mechanical Engineering,* vol 0. No 0., pp. 1 - 11, September 2017.

[22] H. Hamdi, H. Zahouani and J. Bergheau, “Residual stresses computation in a grinding process”, Journal of Materials Processing Technology, vol 147 no 3, pp. 277-285, 20 April 2004.

[23] M. Mahdi and L. Zhang, “Applied mechanics in grinding. Part 7: residual stresses induced by the full coupling of mechanical deformation, thermal deformation and phase transformation.”, *International Journal of Machine Tools & manufacture,* vol 39, pp. 1285 -1298, 1999.

[25] H.K. Tonshoff, J. Peters, I. Inasaki and T. Paul, “Modelling and Simulation of Grinding Processes”, CIRP Annals, vol 41 no 2, pp. 677 – 688, 1992.

[26] X. Chen, W.B. Rowe and D.F. McCormack, “Analysis of the transitional temperature for tensile residual stress in grinding”, Journal of Materials Processing Technology, vol 107 no 1-3, pp. 216-220, 2000.

[27] M.J. Balart, A. Bouzina, L. Edwards and M.E. Fitzpatrick, “The onset of tensile residual stresses in grinding of hardened steels”, Materials Science and Engineering A, vol 367, pp. 132-142, 2004.

[28] M. B. Bhusan, ‘A diagnostic Tool For In-Process Monitoring of Grinding’, Department of Mechanical Engineering, Indian Institute of Technology Madras, Chennai, India, 2012.

[29] X.L Shi, S.C Xiu and H.L Su, “Residual stress model of pre-stressed dry grinding considering coupling of thermal, stress, and phase transformation”, *Advanced Manufacturing,* vol 7, pp. 401 - 410, 31 January 2019.

[30] B. W. Kruszynski, R. Wojcik, “Residual Stress in Grinding”, *Journal of Materials Processing Technology,* vol 109, pp. 254-257, 2001.

[31] S. Mirifar, M. Kadivar and B. Azarhoushang, “First Steps through Intelligent Grinding Using Machine Learning via Integrated Acoustic Emission Sensors”, Journal of Manufacturing and Materials Processing, vol 4 no 35, 25 April 2020.

[32] I. Kalaszi, “Some Remarks on Grinding Wheel Wear and Wheel Life”, Department of Production Engineering, Techincal University, Budapest, Hungary, April 17 1971.

[33] M.Z. Nuawi, M.J.M. Nor, F. Lamin, S. Abdullah and C.K.E. Nizwan, “Tool Life Monitoring using Coefficient of Integrated Kurtosis-based Algorithm for Z-filter (I-kaz) Technique”, Proceedings of the 7th WSEAS International Conference on Wavelet Analysis & Multirate Systems, pp. 147, 2007.

[34] R. Cai and M.N. Morgan, “Intelligent Grinding Database – System Development”, Liverpool John Moores University, Liverpool, United Kingdom.

[35] W.B. Rowe, Y. Chen, J.L. Moruzzi and B. Mills, “A generic intelligent control system for grinding”, Computer Integrated Manufacturing Systems, vol 10 no 3, pp. 231 – 241, 1997.

[36] Y. Zhao, J. Webster and H. Kalister, “In-process size and roundness measuring system for the adaptive control of cylindrical grinding”, *Proceedings of 27th International MTDR Conference,* pp. 289 - 297, 1988.

[37] Y. Inada, “Studies in ultra high speed grinding”, PhD thesis, Tohoku University, Japan, 1996.

[38] W.B. Rowe, “Principles of Modern Grinding Technology”, 1st edition, Oxford, Elsevier, 2009, pp. 23-26.

[39] T.L. Becker Jr., R.M. Cannon and R.O. Ritchie, “An approximate method for residual stress calculations in functionally graded materials”, *Mechanics of Materials,* vol 32, pp. 85 - 97, 2000.

[40] W.B. Rowe, ‘Principles of Modern Grinding Technology’, 1st edition, Oxford, Elsevier, 2009, pp. 292-294.

[41] P. Tan, F. Shen, B. Li and K. Zhou, “A thermo-metallurgical-mechanical model for selective laser melting of Ti6Al4V”, Materials and Design, vol 168, 2019 pp. 1-13.

[42] “Titanium Ti-6Al-4V (Grade 5), Annealed Bar”. Undated. Accessed on: 5 August 2020. [Online]. Available: http://www.matweb.com/search/DataSheet.aspx?MatGUID=10d463eb3d3d4ff48fc57e0ad1037434&ckck=1

[43] A. Gasagara, W. Jin and A. Uwimbabazi, “Modeling of Vibration Condition in Flat Surface Grinding Process”, Shock and Vibration, vol. 2020, pp. 1-2, 2020.

[44] T. Shrestha, “Microstructural evolution, its effect on Ti6Al4V alloy”, October 11, 2019, Thermal Processing. [Online] Available: https://thermalprocessing.com/microstructural-evolution-its-effect-on-ti6al4v-alloy/ Accessed: 20 October 2020.

[45] M.I. Ullah, “Measure of Kurtosis”, July 2 2012, ITFeature, [Online]. Available: https://itfeature.com/statistics/measure-of-dispersion/measure-of-kurtosis) Accessed: 20 October 2020.

[46] “Numerical investigation and an effective predicting system on the Selective Laser Melting (SLM) process with Ti6Al4V alloy - Scientific Figure on ResearchGate”. Accessed on: 15 October 2020. [Online] Available: https://www.researchgate.net/figure/Temperature-dependent-thermal-properties-of-Ti6Al4V-8_tbl1_327731189

[47] T.A. Stolarski, “Tribology in Machine Design”, Oxford, Butterworth-Heinemann, 1990, pp. 20 – 21.

9 Appendices

9.1 MATLab Model Code

Initial Grinding Parameters (Depth of cut varies from 0 to 30)

```
vw = (10000)/(1000*60); %feedrate to be changed as desired
vs = 30; %grinding wheel speed to be changed as desired
ae = linspace(0,30*10^-6,30); %depth of cut
ds = 0.2; %diameter of wheel
b = 0.02; %width of cut
Q = ae.*vw.*b; %MRR
```

Mesh Size and Corresponding Cutting Edge Density (Assume Mesh #100)

```
M = 100; %Mesh Size which varies accordingly
dg = 15.2/M;
C = 1/(2.25*(dg^2)); %Cutting Edge Density
r = 10; %ratio of chip width to thickness
```

Material Properties Required to calculate beta

Note that as temperature increases, beta does not vary much and that is why it is a useful parameter in empircal analysis

```
thermalc = 6.7;
density = 4430;
sheatcap = 526.3;
beta = sqrt(thermalc*density*sheatcap);
```

Assumed Grinding Coefficients for the process (CBN on Ti6Al4V)

```
Cp = 9.5*10^9; %specific grinding energy
tana = 2.34; %a = included angle of grain
mu = 0.3; %grinding coefficient of friction
```

Estimating temperature as a function of parameters stated above (independent of force)

```
thetathree = 25+(1.13*Cp*(vw^0.5)*(ae.^0.75))/(beta*ds^0.25); %Proven accurate
empiral temperature model, used for prediction here
```

Approximating the percentage transition (using Castro's model) for each depth of cut and therefore its apparent work hardness

Assumed that the workpiece is acquired in a 10%Beta phase initially

```
fbeta = zeros(30,1);
betaf = 980;
```

```
transtempalphabeta = 883;
Workhardness = zeros(30,1);
percentagebeta = 0.1 + zeros(30,1);


for n = 1: length(ae)
    if (25<thetathree(n)) && (thetathree(n)<=betaf)
        fbeta(n) = 0.075 + 0.92*exp(-0.0085*(betaf-thetathree(n)));
    elseif thetathree>betaf
        fbeta(n) = 1;
    end

    if fbeta(n)>percentagebeta(n) %statement checks whether beta induced from
grinding is more than initial
        percentagebeta(n) = fbeta(n);
    end
    alphahardness = 300; %fully alpha BHN
    betahardness = 379; %fully beta BHN
    Workhardness(n) = alphahardness + percentagebeta(n)*(betahardness-
alphahardness);
end
```

Determining a K-Factor for to increase the calculated force by based on percentage transition, yield factor and hardness factor.

```
alphayield = 7.6;
betayield = 72.3;
yieldfactor = betayield/alphayield;


hardfactor = (Workhardness/alphahardness)';


forcefact = ((yieldfactor-1)*percentagebeta + 1)';
Fn = ae.*forcefact.*hardfactor*Cp*((pi*vw*b)/(2*vs))*tana;
plot(ae*10^6,Fn);
grid('on');
vwmm = vw*(1000*60);
title({['Normal Grinding Force vs. Depth of Cut'], ['vw = ' num2str(vwmm)
'(mm/min) and vs = ' num2str(vs) '(m/s)']});
xlabel('Depth of Cut');
ylabel('Normal Force (N)');


[aex, thetax] = meshgrid(ae,thetathree);
Fnz = meshgrid(Fn);
surf(aex*10^6,thetax,Fnz);


xlabel('Depth of Cut');
ylabel('Temperature');
zlabel('Normal Grinding Force');
```

```
title('RSM - DepthCut, Temp, NormalGrindingForce');
```

Calculating Other Associated Values

```
Ft = Cp*(vw*ae.*b/vs) + mu*Fn;
P = Ft*vs;
Pw = 0.65*P;
CoeffB = (P/(b*vw))*10^-6;


lc = sqrt(ae.*ds); %contact length
chipthick = sqrt((ae.*vw)/(C*r*lc.*vs)); %chip thickness
q = vs/vw; %speed ratio
c = chipthick^2;
Ra = (c./ae).*(q/(q+1)); %surface roughness
```

Determining Residual Stress and Comparing it to Coefficient B by a plot

```
cte = 9.15*10^-6;
percent = (1-percentagebeta)';
poissonsr = 0.342;
resstress1 = hardfactor.*((114.7*10^9)*cte*(thetathree-25))/(1-poissonsr);
[aex1,tempy] = meshgrid(ae,thetathree);
ResidualStress = ((percent.*114.7*10^9)*cte.*tempy)/(1-poissonsr); %Residual
Stress in MPa %Residual Stress in MPa


surf(aex1*10^6,tempy,ResidualStress*10^-6);
xlabel('Depth of Cut (microns)');
ylabel('Temperature (Celsius)');
zlabel('Residual Stress (MPa)');
title('RSM - DepthCut, Temp, ResStress');


plot(resstress1*10^-6, CoeffB);
title({['Residual Stress vs. Coefficient B'], ['vw = ' num2str(vwmm) '(mm/min)
and vs = ' num2str(vs) '(m/s)']});
xlabel('Residual Stress (MPa)');
ylabel('Coefficient B (unitless)');


plot(ae*10^6,resstress1*10^-6);
title({['Residual Stress vs. Depth of Cut'], ['vw = ' num2str(vwmm) '(mm/min)
and vs = ' num2str(vs) '(m/s)']});
xlabel('Depth of Cut (microns)');
ylabel('Residual Stress');
```

Finding the G Ratio and the Volume of Wheel Removed for a 100mm work length (using the tribology formulae)

```
lengthwork = 0.1; %100mm example
Volworkrem = ae.*b*lengthwork;
H = (Workhardness.*9.81*(10^6))';
W = Fn;
L = lengthwork;
theta = deg2rad(30); %30 degrees;
Volwheelrem = (2/pi())*(tan(theta)./H).*W.*L; %from tribology abrasive wear
G = Volworkrem/Volwheelrem %grinding ratio
```

Creating a table with associated results

```
T = table(ae'*10^6,Fn',Ft',thetathree',resstress1'*10^-
6,CoeffB',Volwheelrem'*10^9,Ra');
T.Properties.VariableNames = {'Depth of Cut (microns)', 'Normal Force (N)',
'Tangen Force (N)', 'Temperature (°C)', 'RStress (MPa)', 'Coeff. B',
'VolWorkRemoved (mm^3)','SurfRough'}
```

9.2 MATLAB Code for Noise Elimination

```
            load("C:\Users\Quintin de
Jongh\Desktop\IGSFullApp\CurrentGrindingData\Force_Data.mat","Data1_Grinding_F
orce","Data1_time_Grinding_Force");

            timeset = Data1_time_Grinding_Force;
            calibrationfactor = 10;
            forceset = Data1_Grinding_Force*calibrationfactor


            fdenoise = wdenoise(double(forceset),5, ...
                'Wavelet', 'sym2', ...
                'DenoisingMethod', 'UniversalThreshold', ...
                'ThresholdRule', 'Soft', ...
                'NoiseEstimate', 'LevelIndependent');
            coeff = ones(1,100)/100;
            movavgforce = filter(coeff,1,fdenoise);
            me = mean(movavgforce);
            mov2 = movavgforce-me;

            plot(timeset,forceset);
            title('Noisy Signal')
            xlabel('Time(s)')
            ylabel('Force(N)')
            plot(timeset,movavgforce);
            title('Denoised Signal')
            xlabel('Time (s)')
            ylabel('Force (N)')
            t1 = find(timeset==2.5)
            t2 = find(timeset==2.6)
            ac = mov2(t1:t2)
            z = kurtosis(ac)
```

Differently plotted code for specific sets in the full data:

```
load("Force_Data.mat");
timeset = Data1_time_Grinding_Force;
forceset = Data1_Grinding_Force;
forceset2 = double(forceset);
combined = [timeset,forceset];
plot(timeset,forceset);
xlim([46.6,46.85])
```

```
a = find(timeset==46.6);
b = find(timeset==46.85);
portio = forceset2(a:b);
```

```
portio1 = wdenoise(portio,5, ...
    'Wavelet', 'sym2', ...
    'DenoisingMethod', 'UniversalThreshold', ...
    'ThresholdRule', 'Soft', ...
```

```
    'NoiseEstimate', 'LevelIndependent');
plot(timeset(a:b),portio1,'r');
coeff = ones(1,100)/100;
avg1secforce = filter(coeff,1,portio1);
plot(timeset(a:b),avg1secforce,'r--');
xlim([46.62,46.85]);
```

9.3 MATLAB System Database Updates, Retrievals, Inserts and Deletes

```
app.startpage = startpage;

            conn = database('IntelligentDatabase','','');
            sqlfind = 'SELECT MaterialName FROM MaterialData';
            data = select(conn,sqlfind);
            arr = table2array(data);
            app.MaterialDropDown.Items = arr;

            sqlfind2 = 'SELECT Specification FROM WheelData';
            data2 = select(conn,sqlfind2);
            arr2 = table2array(data2);
            app.WheelChoiceDropDown.Items = arr2;

            sqlfind3 = 'SELECT CoolantName FROM CoolantData';
            data3 = select(conn,sqlfind3);
            arr3 = table2array(data3);
            app.CoolantDropDown.Items = arr3;
```

```
            batch = app.startpage{1,1}; %integer
            part = app.startpage{1,2}; %integer
            operator = app.startpage{1,3}; %name

            wheel = app.wheelworkcoolant{1,1}; %name
            conn = database('IntelligentDatabase','','');
            sqlf = "SELECT * FROM WheelData where Specification = '";
            sqlfwheel = strcat(sqlf, wheel, "'");
            datawh = select(conn,sqlfwheel);
            widthofcut = (datawh.("Width"))*10^-3;
            diameter = (datawh.("Diameter"))*10^-3;
            meshsize = datawh.("MeshSize");
            grainangle = datawh.("GrainAngle");

            workmat = app.wheelworkcoolant{1,2}; %name
            sqlf = "SELECT * FROM MaterialData where MaterialName = '";
            sqlfwheel = strcat(sqlf, workmat, "'");
            datawork = select(conn,sqlfwheel);
            thermcon = datawork.("ThermalConductivity");
            density = datawork.("Density");
            spheat = datawork.("SpecificHeatCapacity");
            cte = (datawork.("CTE"))*10^-6;
            youngsm = (datawork.("YoungsModulus"))*10^9;
            begtrans = datawork.("BeginTransitionTemp");
            endtrans = datawork.("EndTransitionTemp");
            hardnessone = datawork.("HardnessOne");
            hardnesstwo = datawork.("HardnessTwo");
            yieldone = datawork.("YieldOne");
            yieldtwo = datawork.("YieldTwo");
            burntemp = datawork.("BurnTemp");
            maxresid = datawork.("MaxResidualStress");
```

```
            poisra = datawork.("PoissRatio");


            coolant = app.wheelworkcoolant{1,6}; %name
            sqlf = "SELECT * FROM CoolantData where CoolantName = '";
            sqlfcoolant = strcat(sqlf, coolant, "'");
            datacool = select(conn,sqlfcoolant);
            denscool = datacool.("Density");
            sphcool = datacool.("SpecificHeat");
            thccool = datacool.("ThermalConductivity");
            visccool = datacool.("Viscosity");
```

```
            conn2 = database('IntelligentDatabase','','');
            workmat = app.wheelworkcoolant{1,2};
            length = (app.wheelworkcoolant{1,3});
            width = (app.wheelworkcoolant{1,4});
            transper = (app.wheelworkcoolant{1,5})/100; %convert
            wheel = app.wheelworkcoolant{1,1};
            coolant = app.wheelworkcoolant{1,6};
            flowrate = app.wheelworkcoolant{1,7}; %already in SI

            workspeed = (app.gparams{1,1});
            wheelspeed = (app.gparams{1,2});
            depthcut = (app.gparams{1,3});
            coeffriction = (app.gparams{1,5}); %already in needed untis
            spgenergy = (app.gparams{1,6}); %convert to J/m^3
            gangle = (app.gparams{1,7});

            activetable =
{workmat,length,width,transper,wheel,coolant,flowrate,workspeed,wheelspeed,dep
thcut,coeffriction,spgenergy,gangle};
            colnames =
{'WorkMaterial','WorkLength','WorkWidth','InitialTransPercent','Wheel','Coolan
t','Flowrate','Workspeed','Wheelspeed','DepthofCut','CoeffofFriction','Specifi
cGEnergy','GrindingAngle'};
            wherecl = 'WHERE ID = 1';
            update(conn2,'ActiveCycleData',colnames,activetable,wherecl);
```

```
            conn2 = database('IntelligentDatabase','','');
            workmat = convertCharsToStrings(app.wheelworkcoolant{1,2});
            length = (app.wheelworkcoolant{1,3});
            width = (app.wheelworkcoolant{1,4});
            transper = (app.wheelworkcoolant{1,5})/100; %convert
            wheel = convertCharsToStrings(app.wheelworkcoolant{1,1});
            coolant = convertCharsToStrings(app.wheelworkcoolant{1,6});
            flowrate = app.wheelworkcoolant{1,7}; %already in SI

            workspeed = (app.gparams{1,1});
            wheelspeed = (app.gparams{1,2});
            depthcut = (app.gparams{1,3});
```

```
            coeffriction = (app.gparams{1,5}); %already in needed untis
            spgenergy = (app.gparams{1,6}); %convert to J/m^3
            gangle = (app.gparams{1,7});
            datain =
table(workmat,length,width,transper,wheel,coolant,flowrate,workspeed,wheelspee
d,depthcut,coeffriction,spgenergy,gangle, ...

'VariableNames',{'WorkMaterial','WorkLength','WorkWidth','InitialTransPercent'
,'Wheel','Coolant','Flowrate','Workspeed','Wheelspeed','DepthofCut','CoeffofFr
iction','SpecificGEnergy','GrindingAngle'});
            sqlwrite(conn2,'ReferenceCycleData',datain);
```

```
             conn2 = database('IntelligentDatabase','','');
            batch = app.startpage{1,1}; %integer
            part = app.startpage{1,2}; %integer
            operator = convertCharsToStrings(app.startpage{1,3}); %name

            sqlf = "SELECT MAX(ID) FROM ReferenceCycleData";


            reft = select(conn2,sqlf);
            app.refno = reft.Expr1000;
            datain =
table(batch,part,operator,app.refno,app.fn,app.ft,app.coeffb,app.resid*10^-
6,app.temp,app.gratio,app.srough,app.bocc,app.stexc, ...

'VariableNames',{'PartNo','BatchNo','Operator','ReferenceCycleNumber','MaxFn',
'MaxFt','MaxCoeffB','MaxResidStress','MaxTemp','GrindingRatio','SurfaceRoughne
ss','BurnOccurred','StressExceeded'});
            sqlwrite(conn2,'IndividualPartRecord',datain);
```

```
            conn = database('IntelligentDatabase','','');
            sqlf = "SELECT * FROM MaterialData where MaterialName = '";
            sqlfwheel = strcat(sqlf, workmat, "'");
            datawork = select(conn,sqlfwheel);

            cte = (datawork.("CTE"))*10^-6;
            youngsm = (datawork.("YoungsModulus"))*10^9;

            burntemp = datawork.("BurnTemp");
            maxresid = datawork.("MaxResidualStress");
            poisra = datawork.("PoissRatio");
            thermcon = datawork.("ThermalConductivity");
            density = datawork.("Density");
            spheat = datawork.("SpecificHeatCapacity");
```

```
             workspeed = (app.gparams{1,1});
            wheelspeed = (app.gparams{1,2});
            depthcut = (app.gparams{1,3});
```

```
            coeffriction = (app.gparams{1,5}); %already in needed untis
            spgenergy = (app.gparams{1,6}); %convert to J/m^3
            gangle = (app.gparams{1,7});
            datain =
table(workmat,length,width,transper,wheel,coolant,flowrate,workspeed,wheelspee
d,depthcut,coeffriction,spgenergy,gangle, ...

'VariableNames',{'WorkMaterial','WorkLength','WorkWidth','InitialTransPercent'
,'Wheel','Coolant','Flowrate','Workspeed','Wheelspeed','DepthofCut','CoeffofFr
iction','SpecificGEnergy','GrindingAngle'});
            sqlwrite(conn2,'OptimisedCycleData',datain);
```

9.4 MATLAB System App Code (Body-Only)

Main Menu

```
methods (Access = private)

        % Button pushed function: StartProcessButton
        function StartProcessButtonPushed(app, event)
            StartingPage();
            closereq();
        end


        % Button pushed function: CloseButton
        function CloseButtonPushed(app, event)
            closereq();
        end


        % Button pushed function: HelpButton
        function HelpButtonPushed(app, event)
            message = {'This is the IGS Main Menu.';'Click "Start Process" to
proceed'; 'A short how to guide is provided by clicking the button'};
            msgbox(message, 'Help', 'help');
        end


        % Button pushed function: AcknowledgmentsButton
        function AcknowledgmentsButtonPushed(app, event)
            message = {'The app was created by Quintin de Jongh';'Greatly
Appreciated Support from Prof. Ramesh Kuppuswamy & Fungai Jani'; 'Created for
the completion of BscEng(MEC) at the University of Cape Town, South Africa'};
            msgbox(message, 'Help', 'help');
        end


        % Button pushed function: GuidetoUseButton
        function GuidetoUseButtonPushed(app, event)
            GuideToUse();
        end
    end
```

Guide to Use

```
methods (Access = private)


        % Button pushed function: CloseButton
        function CloseButtonPushed(app, event)
            closereq();
        end
    end
```

Starting Page

```
properties (Access = public)
        batch = 0;
        part = 0;
        operator;
        startpage;
         % Description
    end



    % Callbacks that handle component events
    methods (Access = private)


        % Button pushed function: NextButton
        function NextButtonPushed(app, event)
            app.batch = app.BatchNumberEditField.Value;
            app.part = app.BatchNumberEditField.Value;
            app.operator = app.OperatorDropDown.Value;

            app.startpage{1} = app.batch;
            app.startpage{2} = app.part;
            app.startpage{3} = app.operator;

            WorkWheelInput(app.startpage);
            closereq();
        end


        % Button pushed function: CloseButton
        function CloseButtonPushed(app, event)
            closereq();
        end


        % Button pushed function: HelpButton
        function HelpButtonPushed(app, event)
            message = {'This is the IGS start page.';'Enter the batch and part
number of the part to be ground as well as the operator.';'Reset will set
values to null';'Press next to proceed to the next step.'};

            msgbox(message, 'Help', 'help');
        end


        % Button pushed function: ResetButton
        function ResetButtonPushed(app, event)
            app.BatchNumberEditField.Value = 0;
            app.PartNumberEditField.Value = 0;
            app.OperatorDropDown.Value = 'Quintin';
        end
    end
```

Work, Wheel, Coolant Input Page

```
properties (Access = public)
startpage;
wwc; % Description
end


% Callbacks that handle component events
methods (Access = private)


% Code that executes after component creation
function startupFcn(app, startpage)
app.startpage = startpage;
conn = database('IntelligentDatabase','','');
            sqlfind = 'SELECT MaterialName FROM MaterialData';
            data = select(conn,sqlfind);
            arr = table2array(data);
            app.MaterialDropDown.Items = arr;

            sqlfind2 = 'SELECT Specification FROM WheelData';
            data2 = select(conn,sqlfind2);
            arr2 = table2array(data2);
            app.WheelChoiceDropDown.Items = arr2;

            sqlfind3 = 'SELECT CoolantName FROM CoolantData';
            data3 = select(conn,sqlfind3);
            arr3 = table2array(data3);
            app.CoolantDropDown.Items = arr3;
        end


        % Button pushed function: NextButton
        function NextButtonPushed(app, event)
            app.wwc{1} = app.WheelChoiceDropDown.Value;

            app.wwc{2} = app.MaterialDropDown.Value;
            app.wwc{3} = app.LengthEditField.Value;
            app.wwc{4} = app.WidthEditField.Value;
            app.wwc{5} = app.InitialPhaseTransitionEditField.Value;

            app.wwc{6} = app.CoolantDropDown.Value;
            app.wwc{7} = app.FlowRateEditField.Value;


            GrindingParametersInput(app.startpage, app.wwc);
            closereq();
        end


        % Button pushed function: HelpButton
        function HelpButtonPushed(app, event)
            message = {'This is the Wheel and Workpiece Input Page.';
```

```
                        'Enter the Wheel to use. If you would like to edit or
add a wheel, please do so in the database and then return here.';
                        'Enter the workpiece material and properties as well
as coolant type and flow rate.';
                        'Reset will set all values to null';
                        'Press next to proceed to the next step.'};

            msgbox(message, 'Help', 'help');
        end


        % Button pushed function: CloseButton
        function CloseButtonPushed(app, event)
            closereq();
        end


        % Button pushed function: ResetButton_2
        function ResetButton_2Pushed(app, event)
            app.LengthEditField.Value = 0;
            app.WidthEditField.Value = 0;
            app.InitialPhaseTransitionEditField.Value = 0;
            app.FlowRateEditField.Value = 0;
        end
    end
```

Grinding Parameters Input Page

```
properties (Access = public)
startpage;
wheelworkcoolant; % Description
ggp;
end



    % Callbacks that handle component events
    methods (Access = private)


        % Code that executes after component creation
        function startupFcn(app, startpage, wheelworkcoolant)
            app.startpage = startpage;
            app.wheelworkcoolant = wheelworkcoolant;

        end


        % Button pushed function: ResetButton
        function ResetButtonPushed(app, event)
            app.WorkSpeedEditField.Value = 0;
            app.WheelSpeedEditField.Value = 0;
            app.DepthofCutEditField.Value = 0;
```

```
            app.DesiredSurfaceRoughnessEditField.Value = 0;
            app.CoefficientofFrictionEditField = 0;
            app.SpecificGrindingEnergyEditField = 0;
            app.GrindingAngleEditField = 0;
            app.ButtonGroup.SelectedObject =
app.NoneBurnandStressMinimizedButton;

        end


        % Button pushed function: CloseButton
        function CloseButtonPushed(app, event)
            closereq();
        end


        % Button pushed function: HelpButton
        function HelpButtonPushed(app, event)
            message = {'This is the Grinding Parameters and Properties Input
Page.';
                       'Enter the Values for Work and Wheel Speed as well as
Depth of Cut and Desired Surface Roughness';
                       'Enter the values to the right. Click underneath for
Typical Values, if unsure';
                       'Reset will set all values to null';
                       'Press next to proceed to the next step.'};

            msgbox(message, 'Help', 'help');
        end


        % Button pushed function: NextButton
        function NextButtonPushed(app, event)
            app.ggp{1} = app.WorkSpeedEditField.Value;
            app.ggp{2} = app.WheelSpeedEditField.Value;
            app.ggp{3} = app.DepthofCutEditField.Value;
            app.ggp{4} = app.DesiredSurfaceRoughnessEditField.Value;

            app.ggp{5} = app.CoefficientofFrictionEditField.Value;
            app.ggp{6} = app.SpecificGrindingEnergyEditField.Value;
            app.ggp{7} = app.GrindingAngleEditField.Value;

            app.ggp{8} = app.ButtonGroup.SelectedObject.Value;
            disp(app.ggp{8});

            OutputPage(app.startpage, app.wheelworkcoolant, app.ggp);
            closereq();

        end


        % Button pushed function: TypicalValuesButton
        function TypicalValuesButtonPushed(app, event)
            TypicalValues();
```

```
        end
    end
```

Typical Values Page

```
methods (Access = private)


        % Button pushed function: CloseButton
        function CloseButtonPushed(app, event)
            closereq();
        end
    end
```

Output Page

```
properties (Access = public)
startpage;
wheelworkcoolant;
gparams; % Description
temp = 0;
fn = 0;
ft = 0;
coeffb = 0;
resid = 0;
gratio = 0;
srough = 0;
bocc = "No";
stexc = "No";
refno = 0;
end
methods (Access = public)
function defineproperties(app)
batch = app.startpage{1,1}; %integer
part = app.startpage{1,2}; %integer
operator = app.startpage{1,3}; %name
wheel = app.wheelworkcoolant{1,1}; %name
conn = database('IntelligentDatabase','','');
sqlf = "SELECT * FROM WheelData where Specification = '";
sqlfwheel = strcat(sqlf, wheel, "'");
datawh = select(conn,sqlfwheel);
widthofcut = (datawh.("Width"))*10^-3;
diameter = (datawh.("Diameter"))*10^-3;
meshsize = datawh.("MeshSize");
grainangle = datawh.("GrainAngle");
workmat = app.wheelworkcoolant{1,2}; %name
sqlf = "SELECT * FROM MaterialData where MaterialName = '";
sqlfwheel = strcat(sqlf, workmat, "'");
datawork = select(conn,sqlfwheel);
thermcon = datawork.("ThermalConductivity");
density = datawork.("Density");
spheat = datawork.("SpecificHeatCapacity");
cte = (datawork.("CTE"))*10^-6;
```

```
youngsm = (datawork.("YoungsModulus"))*10^9;
begtrans = datawork.("BeginTransitionTemp");
endtrans = datawork.("EndTransitionTemp");
hardnessone = datawork.("HardnessOne");
hardnesstwo = datawork.("HardnessTwo");
yieldone = datawork.("YieldOne");
yieldtwo = datawork.("YieldTwo");
burntemp = datawork.("BurnTemp");
maxresid = datawork.("MaxResidualStress");
poisra = datawork.("PoissRatio");
coolant = app.wheelworkcoolant{1,6}; %name
sqlf = "SELECT * FROM CoolantData where CoolantName = '";
sqlfcoolant = strcat(sqlf, coolant, "'");
datacool = select(conn,sqlfcoolant);
denscool = datacool.("Density");
sphcool = datacool.("SpecificHeat");
thccool = datacool.("ThermalConductivity");
visccool = datacool.("Viscosity");
length = (app.wheelworkcoolant{1,3})*10^-3; %convert to m
width = (app.wheelworkcoolant{1,4})*10^-3; %convert to m
transper = (app.wheelworkcoolant{1,5})/100; %convert


flowrate = app.wheelworkcoolant{1,7}; %already in SI
workspeed = (app.gparams{1,1})/(1000*60); %convert to m/s
wheelspeed = (app.gparams{1,2}); %in SI already
depthcut = (app.gparams{1,3})*10^-6; %convert to m
desiredroughness = (app.gparams{1,4})*10^-6; %convert to m
coeffriction = (app.gparams{1,5}); %already in needed untis
spgenergy = (app.gparams{1,6})*10^9; %convert to J/m^3
gangle = deg2rad(app.gparams{1,7}); %convert to radians
Q = depthcut*workspeed*widthofcut;
dg = 15.2/meshsize;
C = 1/(2.25*(dg^2));
r = 10;
tana = tan(grainangle);
beta = sqrt(thermcon*density*spheat);
thetathree =
25+(1.13*spgenergy*(workspeed^0.5)*(depthcut.^0.75))/(beta*diameter^0.25);
fbeta = 0;
workhardness = 0;
pbeta = transper;
if ((25<thetathree) && (thetathree<= endtrans))
fbeta = 0.075 + 0.92*exp(-0.0085*(endtrans-thetathree));
elseif thetathree> endtrans
fbeta = 1;
end
if fbeta > pbeta
pbeta = fbeta;
end
workhardness = hardnessone + pbeta*(hardnesstwo-hardnessone);
yieldfact = yieldone/yieldtwo;
hardfact = workhardness/hardnessone;
```

```
forcefact = ((yieldfact-1)*pbeta + 1)';
Fn =
depthcut*forcefact*hardfact*spgenergy*((pi*workspeed*width)/(2*wheelspeed))*ta
na;
Ft = spgenergy*(workspeed*depthcut*width/wheelspeed) + coeffriction*Fn;
P = Ft*wheelspeed;
Pw = 0.65*P;
coefffb = (P/(width*workspeed))*10^-6;
lc = sqrt(depthcut*diameter);
chipthick = sqrt((depthcut*workspeed)/(C*r*lc*wheelspeed));
q = wheelspeed/workspeed;
c = chipthick^2;
Ra = (c/depthcut)*(q/(q+1));
percbinv = 1-pbeta;
ressstress = youngsm*cte*thetathree/(1-poisra);
volworkrem = depthcut*width*length;
H = workhardness*9.81*10^6;
theta = deg2rad(gangle);
volwheelrem = (2/pi)*(tan(theta)/H)*Fn*length;
G = volworkrem/volwheelrem;
vwrem = volwheelrem*10^9;
app.WorkpieceTemperatureEditField.Value = thetathree;
app.ResidualStressEditField.Value = ressstress*10^-6;
app.FinalTransitionEditField.Value = pbeta*100;
app.CoefficientBEditField.Value = coefffb;
app.SurfaceRoughnessEditField.Value = Ra;
app.VolumeofWheelRemovedinOnePassEditField.Value = vwrem;
app.GrindingRatioEditField.Value = G;
if thetathree >= burntemp
app.BurnWillOccurCheckBox.Value = 1;
app.bocc = "Yes";
end
if ressstress*10^-6 >= maxresid
app.OverStressedCheckBox.Value = 1;
app.stexc = "Yes";
end
app.NormalForceEditField.Value = Fn;
app.TangentialForceEditField.Value = Ft;
app.PowerEditField.Value = P;
app.HeatfluxEditField.Value = Pw/(lc*widthofcut);
app.LengthofContactEditField.Value = lc;
app.ContacttimeEditField.Value = lc/workspeed;
app.AreaofContactEditField.Value = lc*widthofcut;
app.ChipthicknessEditField.Value = chipthick;
app.temp = thetathree;
app.fn = Fn;
app.ft = Ft;
app.resid = ressstress;
app.gratio = G;
app.srough = Ra;
app.coeffb = coefffb;
%Updating Recommendations (note this would need to be optimized
%for new values when optimization takes place)
```

```
app.WorkSpeedEditField.Value = workspeed;
app.WheelSpeedEditField.Value = wheelspeed;
app.RecommendedWheelEditField.Value = wheel;
app.RecommendedCoolantEditField.Value = coolant;
app.RecommendedDwellTimeEditField.Value = 0;
end
function updateactive(app)
conn2 = database('IntelligentDatabase','','');
workmat = app.wheelworkcoolant{1,2};
length = (app.wheelworkcoolant{1,3});
width = (app.wheelworkcoolant{1,4});
transper = (app.wheelworkcoolant{1,5})/100; %convert
wheel = app.wheelworkcoolant{1,1};
coolant = app.wheelworkcoolant{1,6};
flowrate = app.wheelworkcoolant{1,7}; %already in SI
workspeed = (app.gparams{1,1});
wheelspeed = (app.gparams{1,2});
depthcut = (app.gparams{1,3});
coeffriction = (app.gparams{1,5}); %already in needed untis
spgenergy = (app.gparams{1,6}); %convert to J/m^3
gangle = (app.gparams{1,7});
activetable =
{workmat,length,width,transper,wheel,coolant,flowrate,workspeed,wheelspeed,dep
thcut,coeffriction,spgenergy,gangle};
colnames =
{'WorkMaterial','WorkLength','WorkWidth','InitialTransPercent','Wheel','Coolan
t','Flowrate','Workspeed','Wheelspeed','DepthofCut','CoeffofFriction','Specifi
cGEnergy','GrindingAngle'};
wherecl = 'WHERE ID = 1';
update(conn2,'ActiveCycleData',colnames,activetable,wherecl);


end
function insertreference(app)
conn2 = database('IntelligentDatabase','','');
workmat = convertCharsToStrings(app.wheelworkcoolant{1,2});
length = (app.wheelworkcoolant{1,3});
width = (app.wheelworkcoolant{1,4});
transper = (app.wheelworkcoolant{1,5})/100; %convert
wheel = convertCharsToStrings(app.wheelworkcoolant{1,1});
coolant = convertCharsToStrings(app.wheelworkcoolant{1,6});
flowrate = app.wheelworkcoolant{1,7}; %already in SI
workspeed = (app.gparams{1,1});
wheelspeed = (app.gparams{1,2});
            depthcut = (app.gparams{1,3});
            coeffriction = (app.gparams{1,5}); %already in needed untis
            spgenergy = (app.gparams{1,6}); %convert to J/m^3
            gangle = (app.gparams{1,7});
            datain =
table(workmat,length,width,transper,wheel,coolant,flowrate,workspeed,wheelspee
d,depthcut,coeffriction,spgenergy,gangle, ...

'VariableNames',{'WorkMaterial','WorkLength','WorkWidth','InitialTransPercent'
```

```
,'Wheel','Coolant','Flowrate','Workspeed','Wheelspeed','DepthofCut','CoeffofFr
iction','SpecificGEnergy','GrindingAngle'});
            sqlwrite(conn2,'ReferenceCycleData',datain);
        end
        function insertpartrecord(app)
            conn2 = database('IntelligentDatabase','','');
            batch = app.startpage{1,1}; %integer
            part = app.startpage{1,2}; %integer
            operator = convertCharsToStrings(app.startpage{1,3}); %name

            sqlf = "SELECT MAX(ID) FROM ReferenceCycleData";


            reft = select(conn2,sqlf);
            app.refno = reft.Expr1000;
            datain =
table(batch,part,operator,app.refno,app.fn,app.ft,app.coeffb,app.resid*10^-
6,app.temp,app.gratio,app.srough,app.bocc,app.stexc, ...

'VariableNames',{'PartNo','BatchNo','Operator','ReferenceCycleNumber','MaxFn',
'MaxFt','MaxCoeffB','MaxResidStress','MaxTemp','GrindingRatio','SurfaceRoughne
ss','BurnOccurred','StressExceeded'});
            sqlwrite(conn2,'IndividualPartRecord',datain);
        end
    end



    % Callbacks that handle component events
    methods (Access = private)


        % Code that executes after component creation
        function startupFcn(app, startpage, wheelworkcoolant, gparams)
            app.startpage = startpage;
            app.wheelworkcoolant = wheelworkcoolant;
            app.gparams = gparams;

            defineproperties(app);
            updateactive(app);
            disp(app.temp);
        end


        % Button pushed function: CloseButton
        function CloseButtonPushed(app, event)
            closereq();
        end


        % Button pushed function: HelpButton
        function HelpButtonPushed(app, event)
            message = {'This is the Output and Recommendations Page';
```

```
                        'Click Open Dewesoft and all inputs will be sent there
as well as a set up RunPage';
                        'Once Done in DeweSoft, click the export button in
Dewesoft, then come back here and click next.'};
            msgbox(message, 'Help', 'help');
        end


        % Button pushed function: NextButton
        function NextButtonPushed(app, event)
            insertreference(app);
            insertpartrecord(app);
            AnalysisPage(app.startpage, app.wheelworkcoolant, app.gparams);
            closereq();
        end
    end
```

Analysis Page

```
properties (Access = public)
startpage;
wheelworkcoolant;
gparams;
% Description
end


% Callbacks that handle component events
methods (Access = private)


% Code that executes after component creation
function startupFcn(app, startpage, wheelworkcoolant, gparams)
app.startpage = startpage;
app.wheelworkcoolant = wheelworkcoolant;
app.gparams = gparams;
load("C:\Users\Quintin de
Jongh\Desktop\IGSFullApp\CurrentGrindingData\Force_Data.mat","Data1_Grinding_F
orce","Data1_time_Grinding_Force");
timeset = Data1_time_Grinding_Force;
calibrationfactor = 10;
forceset = Data1_Grinding_Force*calibrationfactor;


fdenoise = wdenoise(double(forceset),5, ...
'Wavelet', 'sym2', ...
'DenoisingMethod', 'UniversalThreshold', ...
'ThresholdRule', 'Soft', ...
'NoiseEstimate', 'LevelIndependent');
coeff = ones(1,100)/100;
movavgforce = filter(coeff,1,fdenoise);
me = mean(movavgforce);
mov2 = movavgforce-me;
```

```
plot(app.UIAxes,timeset,mov2);
wheelspeed = (gparams{1,2});
workspeed = (gparams{1,1});
powerset = mov2*wheelspeed;
%these values must be acquired
width = 0.02;
depthcut = 9*10^-6;
diameter = 0.2;
bset = powerset/(width*workspeed);
workmat = wheelworkcoolant{1,2}; %name
conn = database('IntelligentDatabase','','');
sqlf = "SELECT * FROM MaterialData where MaterialName = '";
sqlfwheel = strcat(sqlf, workmat, "'");
datawork = select(conn,sqlfwheel);
cte = (datawork.("CTE"))*10^-6;
youngsm = (datawork.("YoungsModulus"))*10^9;
burntemp = datawork.("BurnTemp");
maxresid = datawork.("MaxResidualStress");
poisra = datawork.("PoissRatio");
thermcon = datawork.("ThermalConductivity");
density = datawork.("Density");
spheat = datawork.("SpecificHeatCapacity");
beta = sqrt(thermcon*density*spheat);
lc = sqrt(depthcut*diameter);
heatflux = 2*powerset/(width*lc);
theta = (heatflux/beta)*(sqrt(lc/workspeed));
ressstress = youngsm*cte*theta/(1-poisra);
plot(app.UIAxes2,ressstress,bset);
app.MaxAllowableStress.Value = maxresid;
app.ActualMaxStress.Value = max(ressstress)*10^-6;
plot(app.UIAxes3,timeset,theta);
app.MaxAllowableTemp.Value = burntemp;
app.ActualMaxTemp.Value = max(theta);
%The below needs to be figured out
%[thetaxx,timesetyy] = meshgrid(theta,timeset);
%resszz = youngsm*cte*thetaxx/(1-poisra);
%surf(app.UIAxes4,thetaxx,timesetyy,resszz);
end


% Button pushed function: CloseButton
function CloseButtonPushed(app, event)
closereq();
end


% Button pushed function: SaveCycleasOptimalButton
function SaveCycleasOptimalButtonPushed(app, event)
conn2 = database('IntelligentDatabase','','');
workmat = convertCharsToStrings(app.wheelworkcoolant{1,2});
length = (app.wheelworkcoolant{1,3});
width = (app.wheelworkcoolant{1,4});
transper = (app.wheelworkcoolant{1,5})/100; %convert
wheel = convertCharsToStrings(app.wheelworkcoolant{1,1});
```

```
            coolant = convertCharsToStrings(app.wheelworkcoolant{1,6});
            flowrate = app.wheelworkcoolant{1,7}; %already in SI

            workspeed = (app.gparams{1,1});
            wheelspeed = (app.gparams{1,2});
            depthcut = (app.gparams{1,3});
            coeffriction = (app.gparams{1,5}); %already in needed untis
            spgenergy = (app.gparams{1,6}); %convert to J/m^3
            gangle = (app.gparams{1,7});
            datain =
table(workmat,length,width,transper,wheel,coolant,flowrate,workspeed,wheelspee
d,depthcut,coeffriction,spgenergy,gangle, ...

'VariableNames',{'WorkMaterial','WorkLength','WorkWidth','InitialTransPercent'
,'Wheel','Coolant','Flowrate','Workspeed','Wheelspeed','DepthofCut','CoeffofFr
iction','SpecificGEnergy','GrindingAngle'});
            sqlwrite(conn2,'OptimisedCycleData',datain);
        end


        % Button pushed function: UpdateButton
        function UpdateButtonPushed(app, event)
            t1 = app.Time1EditField.Value;
            t2 = app.Time2EditField.Value;
            load("C:\Users\Quintin de
Jongh\Desktop\IGSFullApp\CurrentGrindingData\Force_Data.mat","Data1_Grinding_F
orce","Data1_time_Grinding_Force");
            timeset = Data1_time_Grinding_Force;
            calibrationfactor = 10;
            forceset = Data1_Grinding_Force*calibrationfactor;


            fdenoise = wdenoise(double(forceset),5, ...
                'Wavelet', 'sym2', ...
                'DenoisingMethod', 'UniversalThreshold', ...
                'ThresholdRule', 'Soft', ...
                'NoiseEstimate', 'LevelIndependent');
            coeff = ones(1,100)/100;
            movavgforce = filter(coeff,1,fdenoise);
            me = mean(movavgforce);
            mov2 = movavgforce-me;

            time1 = find(timeset==t1);
            time2 = find(timeset==t2);
            datarange = mov2(time1:time2);
            kurt = kurtosis(datarange);
            app.KurtosisCoefficientEditField.Value = kurt;

            if kurt < 1
                app.AssociatedWheelWearEditField.Value = "Insignificant Wear";
            end
            if kurt >= 1 && kurt <=2
                app.AssociatedWheelWearEditField.Value = "Low Wear";
            end
```

```
        if kurt > 2 && kurt <=4.5
            app.AssociatedWheelWearEditField.Value = "Moderate Wear";
        end
        if kurt > 4.5
            app.AssociatedWheelWearEditField = "High Wear";
        end
    end
end
```

Risk Assessment Form

- **Risk Assessments are not required for simple workshop activities covered by the Safety Declaration. However permission must be obtained from the Workshop Manager before commencing any activity in the Workshop.**
- **A new Risk Assessment form needs to be completed for each major activity within your project.**
- **You are required to include this document (signed) in your bound project submission and mount a copy next to any rig / apparatus you are using.**

Your Name	Quintin Oliver de Jongh
Your Supervisor	Ramesh Kuppuswamy
Project title and number	Development of Intelligent Grinding System – Project 31
Area Safety Warden	

This Section to be completed by the student (Must be typed and the declaration signed)			
Location where the activity will be done	At the Residence of the Student		
Describe the activity (use attachment with diagrams if needed)	Continuous usage of computer for extended periods of time		
Names of persons involved in this activity	Quintin de Jongh		
Describe in detail the risks you (and others) will face during this activity and the potential consequences of your activities	Damage to eyesight due to viewing bright screens for a long time. Musculoskeletal problems in areas such as back, neck, chest, arms, shoulders and feet. Repetitive strain injuries such as carpal tunnel syndrome. Sleeping problems due to artificial lighting late at night.		
Does this activity involve the use of any materials (chemicals, gasses, etc.) which may be hazardous to health, or the environment	No ✔	Yes	*If Yes list the chemicals / gasses to be used and attach the MSDS(s) for these materials.*
Does this activity involve any equipment / device designed or built by you which is to be plugged into mains electricity?	No ✔	Yes	*If Yes please consult with Mr Richard Whittemore or* Mr Maysam Soltanian (Electrical Machines Lab) *before connecting to the mains / switching on.*
Does the activity involve any new equipment / devices designed by you which will be pressurized with a gas or volatile liquids?	No ✔	Yes	*If Yes please check the relevant SANS Pressurised Equipment Regulations and consult with A/Prof Fuls before testing.*
What precautions are required to protect against the risks detailed above?	Lower the brightness of my computer screen and maintain a moderate distance from the screen. Sit with my back straight and take breaks to stretch Allow wrists to be flexible and once again, take breaks from using wrists and hands repetitively and extensively Allow a break of an hour or two before going to sleep without looking at the computer screen		
Describe the personal protective equipment (PPE) required during this activity – specify in detail.	N/A		
Describe the shutdown procedure in detail.	N/A		
Describe any relevant emergency procedures, e.g. spillage response etc.	N/A		
I declare that I am aware of the risks associated with this activity and will take all necessary steps to mitigate these risks.	**Student Signature**		**Date**
			05-05-2020

I am aware of the student's intended activity, and have provided the necessary guidance, inputs, and oversight.	**Supervisor Signature**	**Date**
	K. Ramesh	12th May'20

This section to be completed by the Area Safety Warden		
Level of supervision required (Please tick relevant block)	A = work may not take place without supervisor/warden present.	
	B = work may not take place without a 2nd party present.	
	C = no specific extra supervision requirements. ✔	
I am satisfied that the student is aware of the risks associated with this activity and grant approval for it to proceed.	**Signature**	**Date**
	K. Ramesh	12th May'20