Vibration behavior in modulated tool path (MTP) turning

A Dissertation Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Ryan William Copenhaver
December 2020

DEDICATION

This dissertation is first and foremost dedicated to my lord and savior Jesus Christ. Without my walk with Christ, my journey through life would be a long and hard path with little guiding light.

I would also like to dedicate this dissertation to my parents and siblings who have supported me throughout my academic career. Their unwavering support is what kept me motivated and focused when times were tough.

ACKNOWLEDGEMENTS

I would like to thank my academic advisor and mentor Dr. Tony Schmitz, for his encouragement, support, and guidance. Dr. Schmitz provided an environment that allowed me to form good research practices and encouraged a curiosity towards research. Through his guidance, my career path is forever changed in ways that would not have been available to me before working with him.

I would also like to thank my committee members, Dr. Suresh Babu, Dr. Brett Compton, and Dr. Scott Smith for their support and guidance while completing this dissertation. I am especially thankful for the flexibility and understanding demonstrated by each committee member during the formulation and carrying out of my dissertation work.

I would also like to thank my colleagues at the University of Tennessee, Knoxville, the University of North Carolina at Charlotte, and those in industry for their support and guidance while working on various aspects of my dissertation project. Their advice and expertise have aided me in ways that I will never be able to fully describe.

ABSTRACT

This project studies the process dynamics and surface finish effects of modulated tool path (MTP) turning. In MTP turning, a small amplitude (typically less than 0.5 mm), low frequency oscillation (typically less than 10 Hz) is superimposed on the feed motion by the machine controller to intentionally segment the traditionally long, continuous chips. The basic science to be examined is the vibration behavior of this special case of interrupted cutting, which is not turning because the chip formation is intentionally discontinuous and is not milling because the time-dependent chip geometry is defined by the oscillatory feed motion, not the trochoidal motion of a rotating and translating milling cutter. The hypothesis that MTP will exhibit forced vibration and secondary Hopf bifurcation (a type of unstable machining conditions) depending on the MTP and machining parameters is tested. A physics-based model of the MTP process is derived and implemented through a second-order, time-delay differential equation math model. This model is used to establish the relationship between: 1) the vibration behavior; and 2) the MTP amplitude and frequency, chip width, spindle speed, nominal feed, and structural dynamics. Experiments are presented to validate the math model accuracy and understand the implications of machining stability and workpiece surface finish.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE
INTRODUCTION

Machining stability

Material removal process, such as turning and milling, make up a large portion of modern manufacturing. The ability to produce accurate parts with a desired dimensional tolerance and required surface finish are essential for manufacturers to produce parts economically. Since the widespread adoption of computer numerical control (CNC) in the 1960s, one of the largest hindrances to manufacturing acceptable parts is the presence of unstable (chatter) cutting behavior. Unstable cutting behavior is typically defined as excessive vibration of either the cutting tool or the workpiece. These self-excited vibrations arise from the regeneration of the surface waviness left behind by the cutter from previous cutting passes. In turning, the waviness regeneration occurs between the workpiece revolutions, while in milling the waviness regeneration occurs between the subsequent teeth on the cutter. Prior research has been devoted to relating the tool/workpiece combination structural dynamics to the stability limit.

Traditionally, cutting behavior has been predicted using an analytical stability solution that provides a stability lobe diagram. An example stability lobe diagram is provided in Figure 1.1. The diagram derives its name from the U-shaped curves, or ‘lobes of stability’, that describe the limit between stable and unstable behavior as a function of spindle speed. In the figure, the shaded region represents predicted unstable cutting while the unshaded region predicts stable cutting.

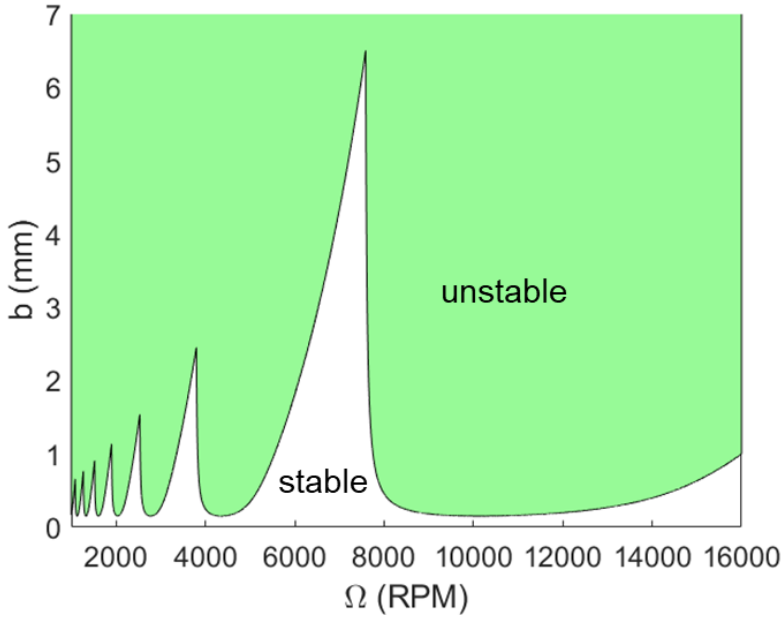

Figure 1.1: Example stability lobe diagram. The shaded region corresponds to predicted unstable cutting behavior. In the figure, b, corresponds to the depth of cut while Ω is the spindle speed.

The presence of unstable cutting (commonly referred to as chatter) may be identified using different methods. These unstable machining behavior identification methods include: 1) recording a representative dynamic signal, such as the audio signal of the cutting process, to identify the presence of a chatter frequency (a frequency other than the fundamental forcing or tooth passing frequency and its multiples). In traditional turning, there is no forcing frequency because the commanded chip thickness remains constant throughout the cut. In milling however, since the cutter typically has multiple teeth and is rotating while translating, the fundamental forcing (tooth passing) frequency is the product of the rotating speed and number of teeth. The audio signal is analyzed using the fast Fourier transform (FFT) and the frequency content is analyzed to detect if any other frequencies (i.e. the chatter frequency) are present. 2) Chatter may also be identified by examining the workpiece surface finish. Using profilometry equipment, microscopes, and other surface finish evaluations tools, the workpiece surface is evaluated for the presence of unstable cutting due to increased surface roughness and other larger spatial wavelength features (namely, induced waviness that corresponds to the rotating frequency of the tool or workpiece and the chatter frequency).

Summary: Unstable cutting behavior results from the regeneration of waves left behind from previous passes in milling and turning. Prior work has been conducted to relate the tool and workpiece dynamics to the cutting stability limit using a stability lobe diagram. Cutting stability has traditionally been analyzed using the dynamic process signals and the workpiece surface.

Modulated Tool Path (MTP) originality

Unlike milling operations where the rotating tool constantly engages and disengages the workpiece to produce intermittent cutting conditions, conventional turning, boring, and threading operations typically exhibit continuous cutting. Once the cutting edge is engaged with the workpiece, it remains in contact at a specified feed rate until the cut concludes. This tends to produce a continuous chip that can wrap and collect near the cutting edge when machining ductile materials; see Figure 1.2. The local buildup of this continuous chip can result in one or more of several undesirable outcomes including workpiece scratching, tool or machine damage, machinist injury, and/or increased cycle time to clear the chip(s) from the work area.

Traditionally, specialized rake face geometries (i.e., chip breakers) have been used with specified depth of cuts, feed rates, and material groups to ensure that discontinuous chips are generated. Another common chip management strategy is to use a high-pressure coolant stream to cause the generated chip to curl and break. These strategies are highly dependent on the machining process parameters [2] . An alternative approach to these techniques is MTP turning, where discrete chips are formed by repeatedly interrupting the continuous chip formation using the machine axes to superimpose low frequency tool oscillations on the nominal tool feed motion. In this case, successful chip separation is based on: 1) the oscillation frequency relative to the spindle speed; and 2) the oscillation amplitude relative to the global feed per revolution. An exaggerated depiction of an MTP tube turning operation is displayed in Figure 1.3.

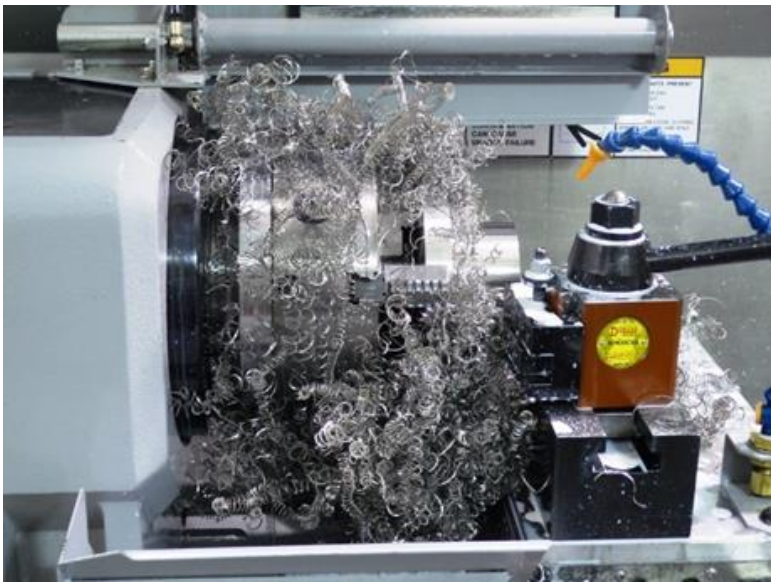

Figure 1.2: Chip buildup in a turning operation.

A sequence of high-speed video images is presented in Figure 1.4 to demonstrate MTP discontinuous chip formation. The testing setup is the same as Figure 1.3. These images show the increase and decrease in chip thickness that occur in each oscillation cycle. In Figure 1.4, the cutting speed is constant, but the chip thickness varies continuously.

The broken chip length is dependent on two, user-defined MTP parameters: 1) the tool (axis) oscillation frequency relative to the spindle speed, or Oscillations Per Revolution (OPR); and 2) the oscillation amplitude relative to the global feed per revolution, or the Ratio of the Amplitude to the Feed rate (RAF). These MTP parameters are defined as: $OPR = \frac{60*f}{\Omega}$ and $RAF = \frac{A}{f_r}$, where *f* is the tool oscillation frequency (Hz) in the feed direction, Ω is the spindle speed (rpm), A is the tool oscillation amplitude (mm), and f_r is the global feed per revolution (mm/rev) for a traditional, constant feed turning operation. In Figure 1.5, an MTP parameter plot with analytical chip breaking boundary is shown. Figure 1.5 is analogous to the stability lobe diagram (shown in Figure 1.1) in that if parameters are chosen in one of the lobes, segmented chips will be generated, whereas other parameter pairs result in continuous chip generation.

The MTP approach is related to previous and current studies of modulation assisted machining (MAM). The differences are that: 1) a separate transducer (typically piezo-based) is used to provide the sinusoidal tool motion in MAM; and 2) the oscillation frequencies are higher in MAM. MTP is limited to (typically) 10 Hz or less depending on the machine tool controller performance (the oscillating

motion is defined in the CNC part program), while MAM frequencies have been reported from tens of Hz to 1 kHz using piezo transducers.

Summary: MTP and MAM are important and innovative additions to advanced machining technology, but a comprehensive process dynamics and surface finish simulation capability has not been presented.

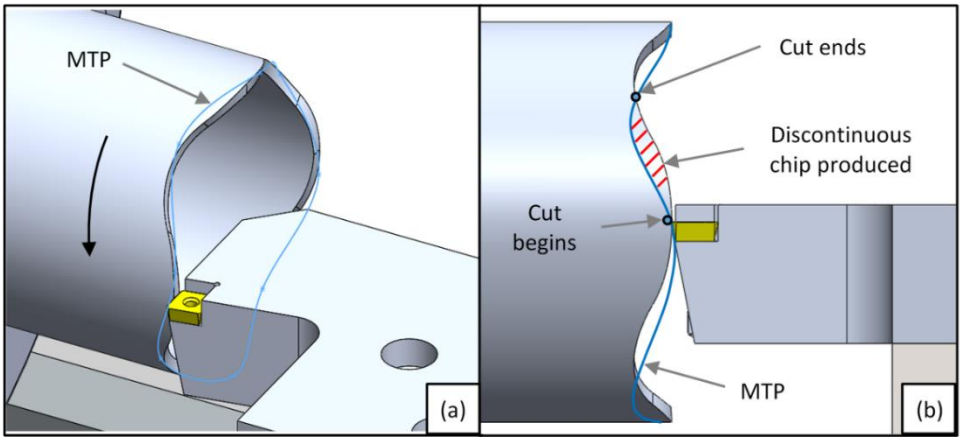

Figure 1.3: (a) The tool feed motion along the tube axis is varied sinusoidally to produce a wavy surface in the feed direction. (b) By selecting appropriate OPR and RAF values, broken chips are periodically produced (amplitude exaggerated).

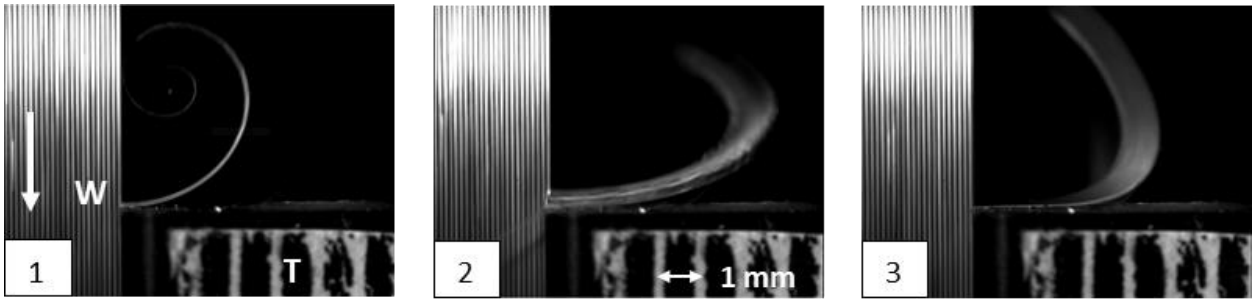

Figure 1.4: High speed video images of workpiece (W) and tool (T). The behavior proceeds from (1) initial chip generation along the tool rake face, to (2) maximum chip thickness, to (3) discrete chip generation.

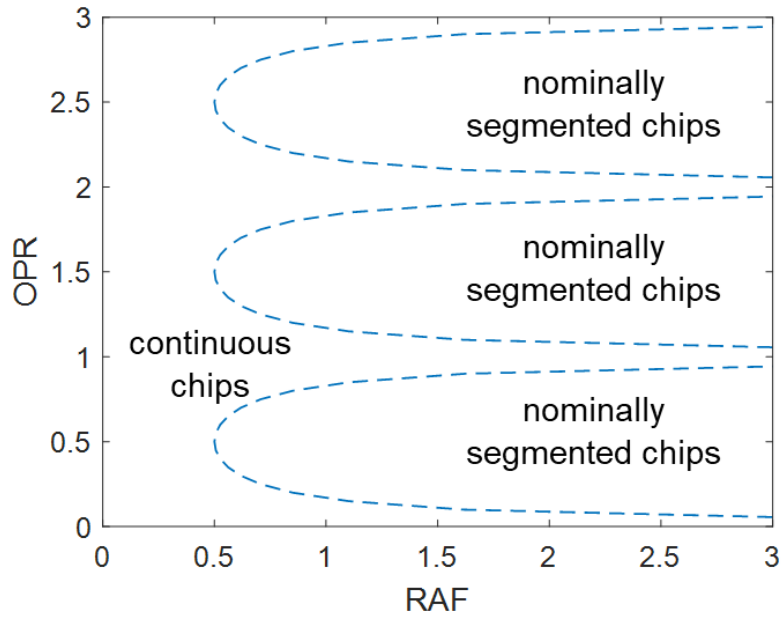

Figure 1.5: The MTP parameter map with the analytical chip breaking boundary (dashed line) is provided.

CHAPTER TWO
LITERATURE REVIEW

The earliest semblance of a lathe appears in Egyptian hieroglyphs as a way to sculpt clay for pots or to shape wooden objects; however, the mass adoption of using a lathe to produce metal parts did not come to fruition until the industrial revolution in the late 1700s [3] . Increased productivity through the implementation of CNC in the 1960s firmly established turning centers as a major material removal process in manufacturing. One issue that has been a constant area of research for turning operations is how to manage the chips that are generated during the cutting process. Zhang [1] conducted research into the effectiveness of specialized rake face geometries (also known as chip breakers). Rasch and Viegeland [2] evaluated the effectiveness of high-pressure coolant directed at the tool-workpiece interface. Both manuscripts stated that the performance of these strategies strongly depends on the chip thickness, chip radius of curvature, and workpiece material, as well as the coolant pressure, direction, and location when high pressure coolant is applied.

An alternative solution to these techniques is MTP turning. MTP turning ensures discrete chips are formed by repeatedly interrupting the continuous chip formation using the machine axes to superimpose low frequency tool oscillations on the nominal tool feed motion. MTP has previously been investigated by several researchers. Adams [4] investigated the effect of the tool path profile shape and planning and its effect on the chip breaking performance. Assaid [5] assessed and described how to program MTP tool paths as well as a method to analyze the machine tool’s capability to perform the commanded moves. Tursky [6] studied

how to interpret the machine tool's capability to perform commanded moves and use that information to select the optimal MTP machining parameters. Tursky also explored the effects on cutting temperature and tool life when machining with MTP compared to traditional turning. Woody et al. [7] conducted additional chip breaking experiments using MTP as well as further cutting temperature and tool life experiments. The results from the experiments suggested that while cutting with MTP the cutting temperature was lower due to the intermittent cutting which lead to greater tool life. A patent for the use of the machine tool axes to break chips using MTP tool paths was obtained by Woody et al. [9] . Smith et al. [10] - [16] further investigated the effects of the machine tool's capability to perform commanded motion. Through the degradation of the oscillatory motion at higher amplitudes and frequencies of vibration, the ability to generate discontinuous chips was lost. The authors also reported the effects of implementing MTP turning to reduce the cutting temperatures when machining difficult to machine materials. McFarland [17] presented a simulation to predict the workpiece surface finish while using MTP turning on a diamond turning machine; good agreement between predicted and measured surface profiles was shown. Berglind and Ziegert [18] - [19] developed a new machine tool characterization method that accounted for both the machine tool's dynamic oscillation capabilities and the offset error experienced when performing a combined axis motion. Using the new machine tool characterization method, an automatic MTP parameter selection for constant surface speed was presented. Additionally, Berglind and Ziegert [20] demonstrated the effectiveness in applying MTP to outer diameter threading

operations of ductile materials through successfully generating discontinuous chips. Rubeo et al. [21] showed initial results of an experimental in-process metrology test bed to capture high-speed micro videography, cutting force, tool-chip temperature, and tool wear measurements during orthogonal turning. Copenhaver et al. [22] conducted AISI 1026 cold-drawn steel machining experiments where data was presented for the feed motion and modeling, force measurement and modeling, temperature measurement, and chip formation for constant and MTP tool paths. Copenhaver et al. [23] introduced a periodic sampling method suitable for determining MTP cutting stability. Stable and unstable cutting while breaking chips with MTP was presented.

Modulation Assisted Machining (MAM) is another chip breaking technique that has been implemented in several machining process to control the formation of chips. MAM employs the use of an external actuator instead of the machine tool's axes to facilitate the superimposed tool path oscillations. Toews, Compton, and Chandrasekar [24] demonstrated MAM's effectiveness in drilling through measuring the cutting force and torque and saw no significant increase when compared to traditional machining tool paths. Moscoso et al. [25] showed that the effectiveness of the machining lubricant increases when machining with MAM due to the increased fluid penetration into the tool-workpiece interface. Mann et al. [26] - [29] demonstrated that different chip geometries are generated (equiaxed, platelet, and fiber-shaped particles) by changing the MAM parameters. The coefficient of friction was also shown to decrease with the proper selection of MAM parameters, which leads to a reduction in the required shearing energy. Guo et al.

[30] - [34] reported a 20× increase in tool life when machining compacted graphite iron (CGI) using MAM compared to traditional machining. An analytical model to predict the chip morphology and surface topography was investigated for different MAM parameters accounting for tool geometries, cutting conditions, modulation conditions, and the effects of plastic side flow. An analytical force and temperature model using the MAM and machining parameters as inputs was compared to orthogonal cutting tests and was shown to have good agreement between prediction and measurement [32] - [34] . Bebnath and Singh [35] showed that when MAM is applied to drilling carbon fiber reinforced plastics (CFRP) the drilled hole quality is increased when compared to conventional drilling.

Summary: The effect of MTP, MAM, and other chip breaking methods on the cutting force, cutting temperature, workpiece surface finish, and chip breaking effectiveness is an active area of research with several researchers involved in the development of this new advanced manufacturing technique.

CHAPTER THREE
MATERIALS AND METHODS: MTP TURNING STABILITY

MTP dynamic simulation

In order to model the tool vibration and cutting force during MTP turning, a time domain simulation has been derived. The simulation details are provided in the following paragraphs.

In each time step of the simulation, the instantaneous chip thickness is calculated by considering the current and all previous surfaces. The cutting force is then calculated using this chip thickness, the chip width, and cutting force model. Once the force is known, the second-order, time-delay differential equations of motion for the flexible cutter are solved by numerical integration. The corresponding tool displacement is then used together with the commanded MTP motion to calculate the chip thickness in the next time step. For numerical integration using the fixed time step (Euler) approach, the requirement is that the time step is small enough to avoid numerical instability. The integration step is typically selected to be *N* times smaller than the smallest vibration period for the structural dynamics. Because the smallest period corresponds to the highest natural frequency, the time step, dt, is specified using Eq. 1:

$$dt = \frac{1}{N \cdot f_{n,max}}, \tag{1}$$

where $f_{n,max}$ is the highest natural frequency for all vibration modes in both the u_1 or u_2 directions for the cutting tool. These directions are identified in Figure 3.1.

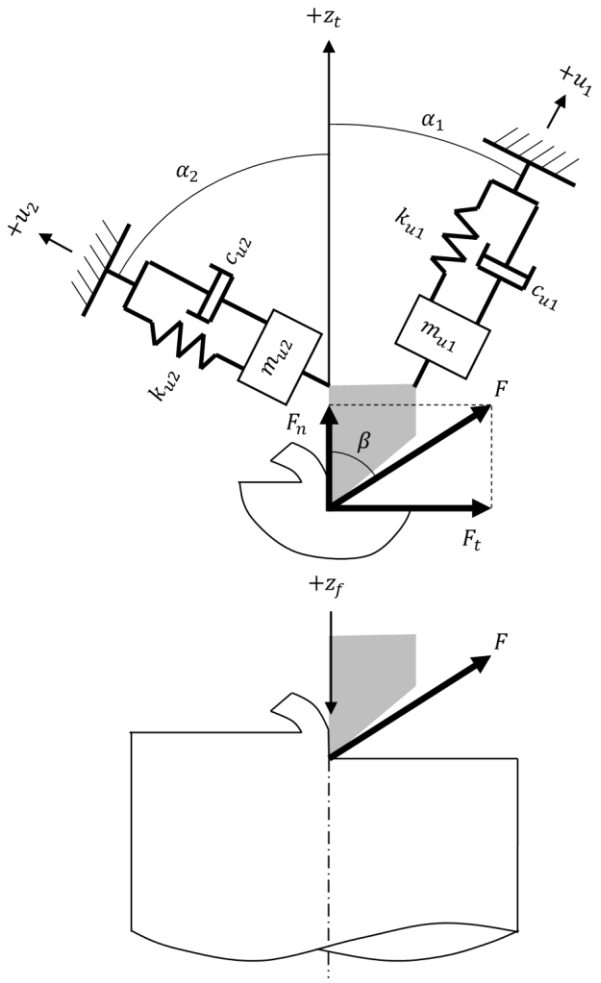

Figure 3.1: (Top) Flexible tool MTP turning dynamics model. The tangential, F_t, normal, F_n, and resultant force, F, components are identified, as well as the modal parameters that represent the structural dynamics in two orthogonal directions, u_1 and u_2. The MTP feed motion, z_f, and tool vibration, z_t, are also identified. (Bottom) Model orientation for turning.

Given the time step, the simulation time vector and corresponding MTP feed motion, z_f, are described. The time vector, t, is defined from zero to the maximum simulation time in equal increments, dt. The MTP feed motion is then specified by Eq. 2:

$$z_f = \left(\frac{\Omega}{60} f_r\right) t + RAF \cdot f_r \cdot \sin\left(\frac{\Omega}{60} 2\pi \cdot OPR \cdot t\right), \quad (2)$$

where Ω is the spindle speed (rpm), f_r is the feed per revolution (mm/rev), RAF is the ratio of the MTP amplitude to the feed per revolution, and OPR is the number of sinusoidal MTP oscillations per revolution of the rotating part. Figure 3.2 displays the MTP feed motion for a spindle speed of 200 rpm, a feed of 0.1 mm/rev, and OPR and RAF values of 0.5 and 0.8, respectively. In the figure, the dashed line denotes the constant feed advance of the tool into the part, while the solid line shows the superposition of the MTP oscillation onto the constant feed. The vertical dotted lines identify each revolution; three revolutions are plotted.

As noted, the first task in each simulation iteration is to calculate the instantaneous chip thickness. Figure 3.3 aids in the calculation description by displaying the Figure 3.2 data parsed by revolution. The revolution numbers are included on the right-hand side of the figure. The nominal chip thickness is the difference between the current tool position and the maximum value of all previous revolutions. Figure 3.3 shows the chip thickness for revolution 2 as the hatched areas. The chip thickness is zero when the revolution 2 oscillation dips below the revolution 1 oscillation. Note that the $+z_f$ direction is positive into the part, so “below” here means away from the part and no cutting occurs.

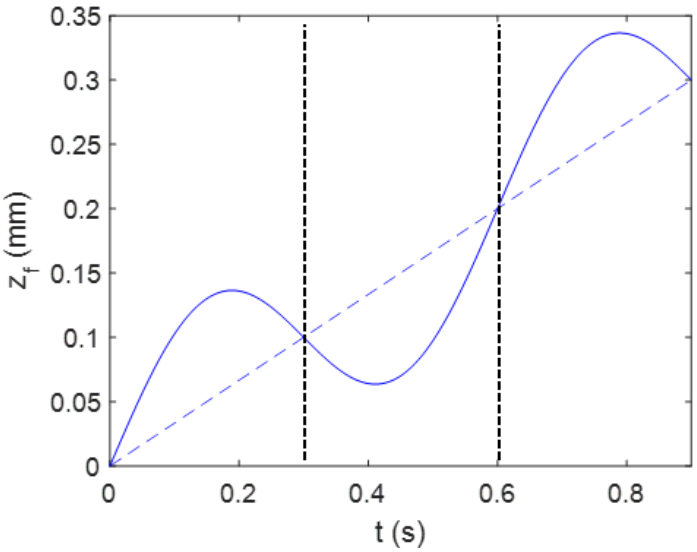

Figure 3.2: MTP feed motion for three spindle revolutions. The spindle speed is 200 rpm, the feed per revolution is 0.1 mm/rev, and the OPR and RAF values are 0.5 and 0.8. The dashed positive slope line identifies the constant feed contribution to the motion, while the solid line shows its sum with the sinusoidal MTP contribution. The vertical dotted lines denote each revolution.

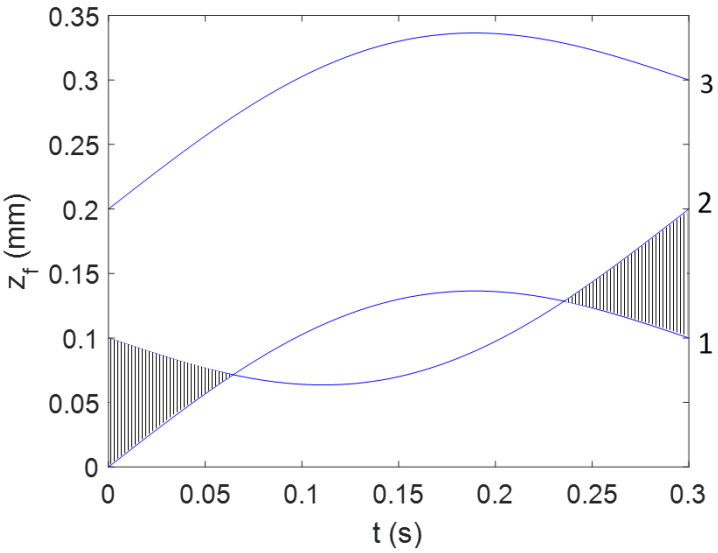

Figure 3.3: Chip thickness calculation for revolution 2. The nonzero chip thickness zones are denoted by the hatched areas.

Figure 3.4 displays the chip thickness for revolution 3. Note that the instantaneous chip thickness is the difference between revolutions 3 and 1 for the time period between 0.0645 s and 0.2355 s and the difference between revolutions 3 and 2 for all other times. The corresponding chip thickness profile for the two revolutions is shown in Figure 3.5. The two revolutions are segmented by the vertical dotted line. Because the OPR is 0.5 for this example, the chip thickness profile in Figure 3.5 repeats every two revolutions in the absence of tool vibrations. MTP turning therefore exhibits periodic excitation, unlike traditional turning where the chip thickness and force are nominally constant.

Figure 3.3 and Figure 3.4 demonstrate the strategy for calculating the instantaneous chip thickness, h. Mathematically, this can be expressed as:

$$h = z_{f,n} - \max\left\{z_{f,n-1}, z_{f,n-2}, \dots\right\}, \tag{3}$$

where n is the current revolution. To include the tool dynamics, which are excited by the periodic forcing function displayed in Figure 3.5, Eq. 3 must be augmented to include the effect of the tool displacement. If z_t is the tool displacement in the surface normal direction and it is considered positive out of the cut (see Figure 3.1), then a positive tool displacement for the current revolution decreases the chip thickness.

A positive tool displacement in a previous revolution, on the other hand, indicates that material that was intended to be removed was left behind. Therefore, a positive tool displacement for the maximum previous revolution yields a larger instantaneous chip thickness in the current revolution.

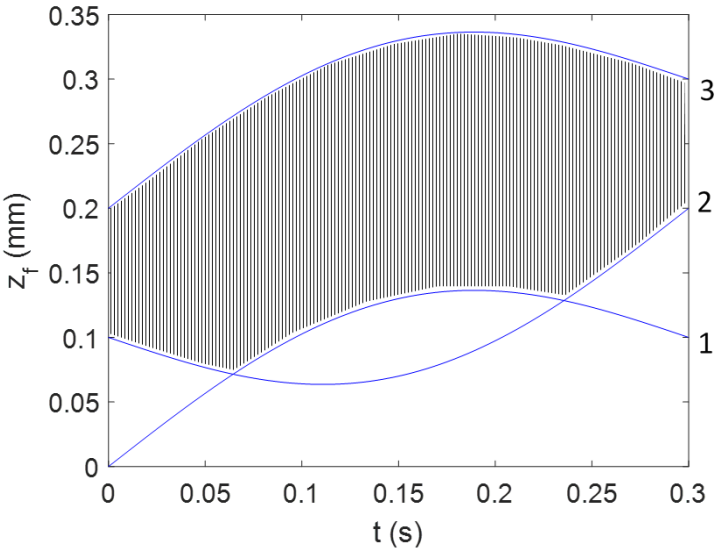

Figure 3.4: Chip thickness calculation for revolution 3. The instantaneous chip thickness is the difference between the current MTP motion and the maximum of all prior revolutions at the same rotation angle.

Equation 3 is now updated to include the tool vibration:

$$h = \left(z_{f,n} - z_{t,n}\right) - \max\left\{\left(z_{f,n-1} - z_{t,n-1}\right), \left(z_{f,n-2} - z_{t,n-2}\right), \ldots\right\}. \quad (4)$$

Returning to Figure 3.1, the chip thickness is now calculated from Eq. 4 as shown in Eq. 5.

$$h = \left(z_{f,2} - z_{t,2}\right) - \left(z_{f,1} - z_{t,1}\right) = z_{f,2} - z_{f,1} - z_{t,2} + z_{t,1} \quad (5)$$

Equation 5 shows the effect of the tool vibrations directly. A positive $z_{t,2}$ reduces the current chip thickness, while a positive $z_{t,1}$ increases the current chip thickness. The final consideration is that Eqs. 3-5 can yield negative values, e.g., during the interval from 0.0645 s and 0.2355 s in Figure 3.3. When $h < 0$, this indicates that no cutting occurs, and the chip thickness is set equal to zero (this introduces a nonlinearity into the system).

Once the chip thickness is known, the resultant cutting force, F, is calculated:

$$F = K_s b h, \quad (6)$$

where K_s is the specific cutting force coefficient and b is the chip width. The resultant force is related to the tangential and normal direction force components through the force angle, β.

$$F_t = F \sin\beta = (K_s \sin\beta) bh = k_t bh \quad (7)$$

$$F_n = F \cos\beta = (K_s \cos\beta) bh = k_n bh \quad (8)$$

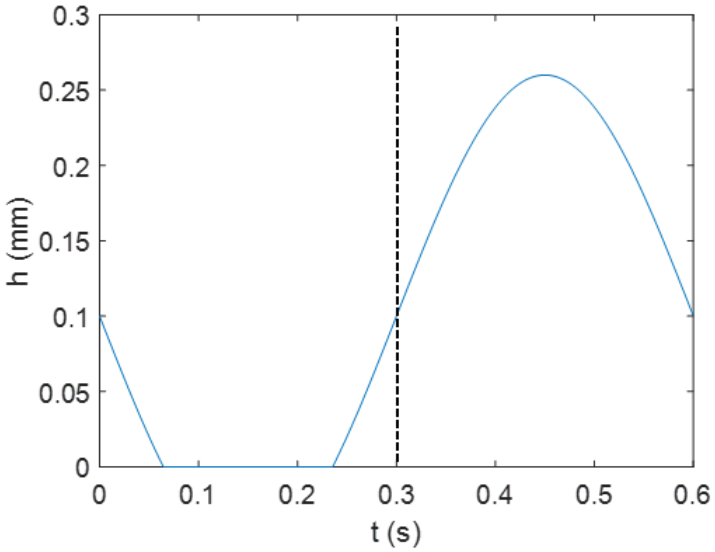

Figure 3.5: Instantaneous chip thickness for revolutions 2 and 3 considering MTP motion only.

Since MTP cutting exhibits a range of chip thicknesses through the commanded process (see Figure 3.5), a power law relationship is implemented. This captures the effect of small instantaneous chip thickness values on the cutting force coefficient value. The revised tangential and normal force models are defined in Eq. 9 and Eq. 10

$$F_t = (a_t h^{b_t} + c_t) bh \tag{9}$$

$$F_n = (a_n h^{b_n} + c_n) bh \tag{10}$$

where a, b, and c denote power law fitting values and the t and n subscript denotes the tangential and normal directions, respectively. The resultant force is projected into the two mode directions to determine the corresponding displacements u_1 and u_2.

$$F_{u1} = F\cos(\beta - \alpha_1) \tag{11}$$

$$F_{u2} = F\cos(\beta + \alpha_2) \tag{12}$$

The Euler integration procedure used to determine the current tool displacement components in the u_1 and u_2 directions proceeds according to Eqs. 13 and 14 [36] .

$$\begin{aligned} \ddot{u}_1 &= \frac{F_{u1} - c_{u1}\dot{u}_1 - k_{u1}u_1}{m_{u1}} \\ \dot{u}_1 &= \dot{u}_1 + \ddot{u}_1 dt \\ u_1 &= u_1 + \dot{u}_1 dt \end{aligned} \tag{13}$$

$$\begin{aligned} \ddot{u}_2 &= \frac{F_{u2} - c_{u2}\dot{u}_2 - k_{u2}u_2}{m_{u2}} \\ \dot{u}_2 &= \dot{u}_2 + \ddot{u}_2 dt \\ u_2 &= u_2 + \dot{u}_2 dt \end{aligned} \tag{14}$$

In Eqs. 13 and 14, m, c, and k are the modal mass, damping, and stiffness values, respectively, and the over-dots indicate time derivatives. The current

acceleration is first calculated using the current force component; recall that current force is determined from the current chip thickness. This acceleration is then used to update the current velocity, where the product of the acceleration and time step is summed with the velocity from the previous time step. The same pattern is used to update the current displacement in each direction. Once the u_1 and u_2 displacements are known, they are projected into the surface normal direction to determine the new tool displacement (Eq. 15). Note that multiple modes in each direction may also be modeled by summing the modal contributions.

$$z_t = u_1 \cos\alpha_1 + u_2 \cos\alpha_2 \tag{15}$$

Stability determination and metric

To establish the MTP turning stability, periodic sampling is implemented as shown in [37] and [23] , where the process signals are sampled at the forcing period. The discretized sampling period, *SP*, is defined in Eq. 16, where *SR* is the number of steps per revolution; see Eq. 17. If the process is stable, the sampled points repeat. If it is unstable, they do not repeat.

$$SP = \frac{SR}{OPR} \tag{16}$$

$$SR = \frac{60}{dt \cdot \Omega} \tag{17}$$

To automatically differentiate between stable (periodic) and unstable (secondary Hopf) conditions, the metric, *M*, is applied to the sampled points:

$$M = \frac{\sum_{i=2}^{N} |z_{ts}(i) - z_{ts}(i-1)|}{N} \tag{18}$$

where z_{ts} is the vector of once-per-MTP period sampled z_t displacements and N is the length of the z_{ts} vector [27] . For stable cuts, the M value is ideally zero (within the limits of numerical precision). For unstable cuts, however, $M > 0$. The use of this metric enables multiple simulations to be completed over a range of OPR and RAF values and a stability map to be automatically produced by plotting a single contour at an arbitrarily small M value. A schematic map of the simulation input-output relationships is provided in Figure 3.6, where the output force, displacement, and velocity signals are simulated and provided to be periodic sampled to identify stable and unstable behavior (i.e., forced vibration or secondary Hopf bifurcation).

Using Figure 1.5, the MTP dynamic simulation can be carried out for a grid of points across the MTP parameter map to generate a MTP stability map. A diagram visualization of the grid of stimulation points on a MTP parameter map is provided in Figure 3.7. The mesh size is determined by the programmer. A finer mesh of simulation points results in higher resolution; however, this also results in larger computation times to generate the map.

Example stability results

To demonstrate the time domain simulation, an example is provided. The simulation specifications are: Ω = 600 rpm, f_r = 0.17 mm/rev, b = 0.8 mm, K_s = 700 N/mm^2, β = 70 deg, α_1 = 90 deg, and α_2 = 0 deg. The modal parameters for the u_1 direction are: k_{u1} = 1×10^7 N/m, m_{u1} = 1.013 kg, and c_{u1} = 318.3 N-s/m, where k is stiffness, m is mass, and c is the viscous damping coefficient.

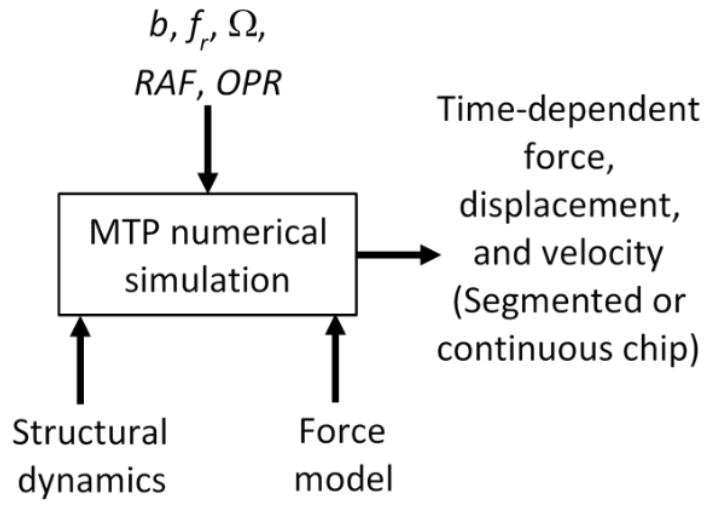

Figure 3.6: Simulation input-output relationships.

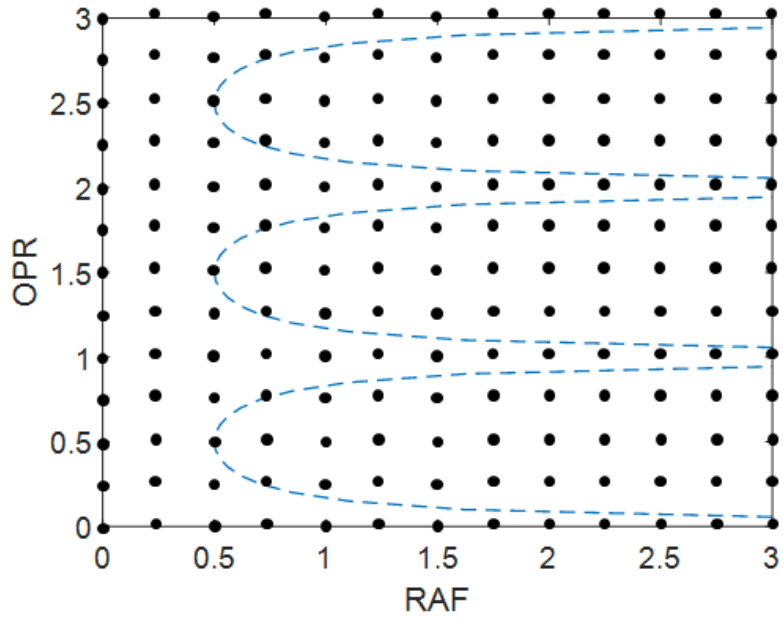

Figure 3.7: Grid of MTP dynamic simulation points overlayed on a MTP parameter plot.

The modal parameters for the u_2 direction, which is described by two vibration modes, are: $k_{u2,1}$ = 5×10^6 N/m, $m_{u2,1}$ = 1.407 kg, and $c_{u2,1}$ = 53.05 N-s/m and $k_{u2,2}$ = 7×10^6 N/m, $m_{u2,2}$ = 0.362 kg, and $c_{u2,2}$ = 159.2 N-s/m. The chip thickness variation for RAF and OPR values of 1.0 and 0.5 is displayed in Figure 3.8. The periodic sampling is shown as well (circles). The tool displacement and force signals are presented in Figure 3.9 and Figure 3.10. A Poincaré map, which plots the displacement versus velocity with periodic sampling, is displayed in Figure 3.11. Because the periodically sampled points repeat (they are superimposed at a single location in Figure 3.11), the cut is stable. Figure 3.12 - Figure 3.15 show the results for the same simulation parameters, but RAF and OPR values of 1.0 and 0.25. These figures demonstrate a secondary Hopf bifurcation (self-excited vibration) and show that MTP parameters affect the process stability.

Summary: A numerical simulation was presented to solve the second-order, time-delay differential equations of motion that describe the MTP (and MAM) process dynamics. The simulation output force and displacement signals were analyzed to identify stable and unstable behavior.

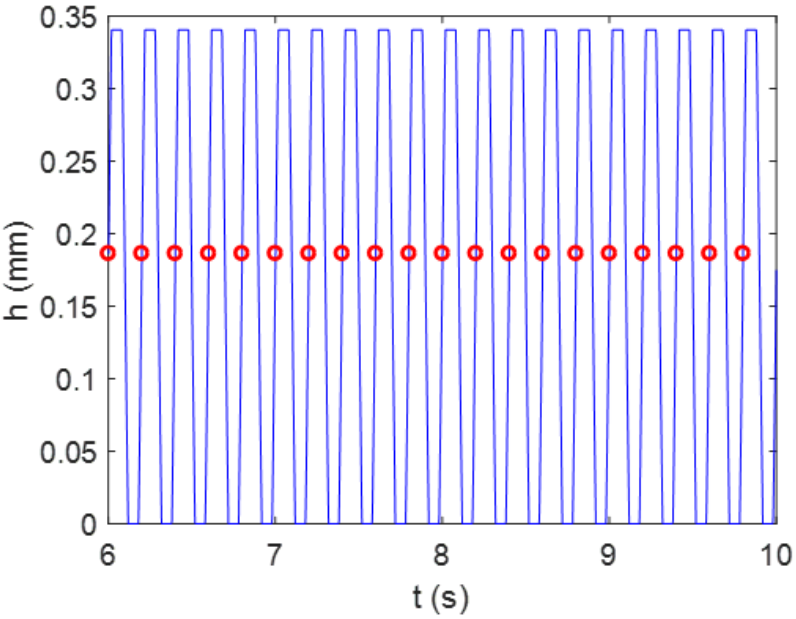

Figure 3.8: Time domain chip thickness for RAF = 1.0 and OPR = 0.5.

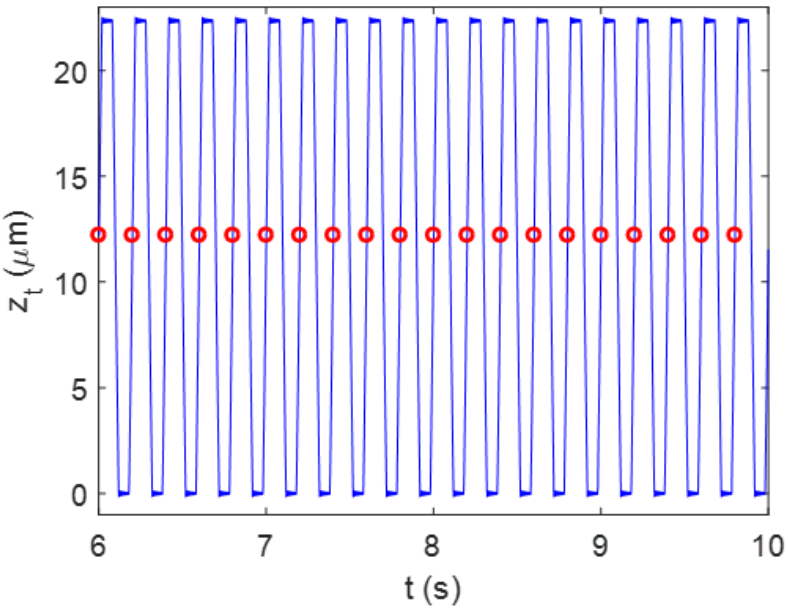

Figure 3.9: Time domain tool displacement for RAF = 1.0 and OPR = 0.5.

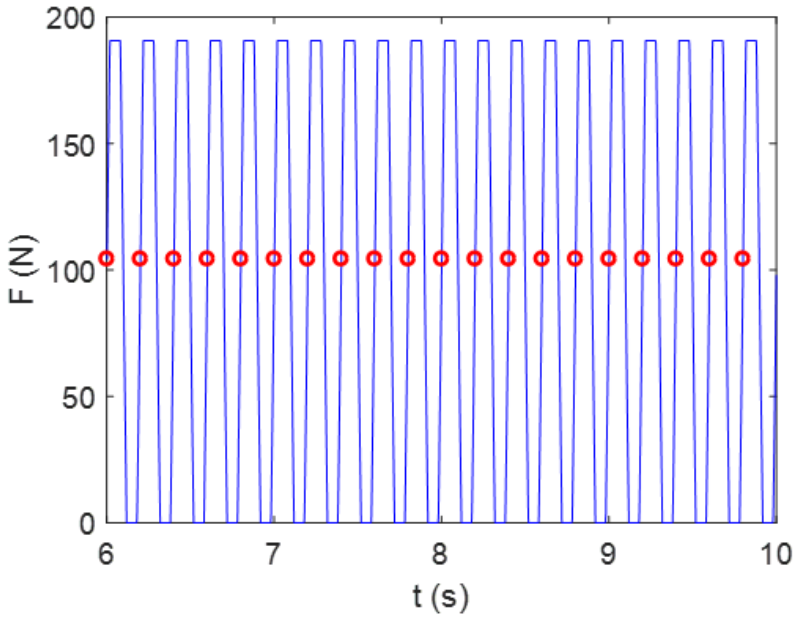

Figure 3.10: Time domain resultant force for RAF = 1.0 and OPR = 0.5.

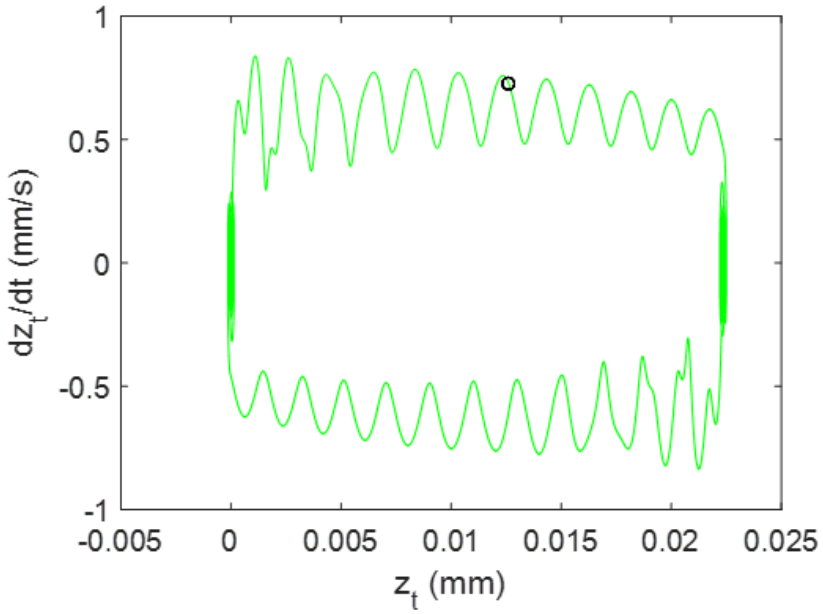

Figure 3.11: Poincaré map of sampled tool displacement versus velocity for RAF = 1.0 and OPR = 0.5.

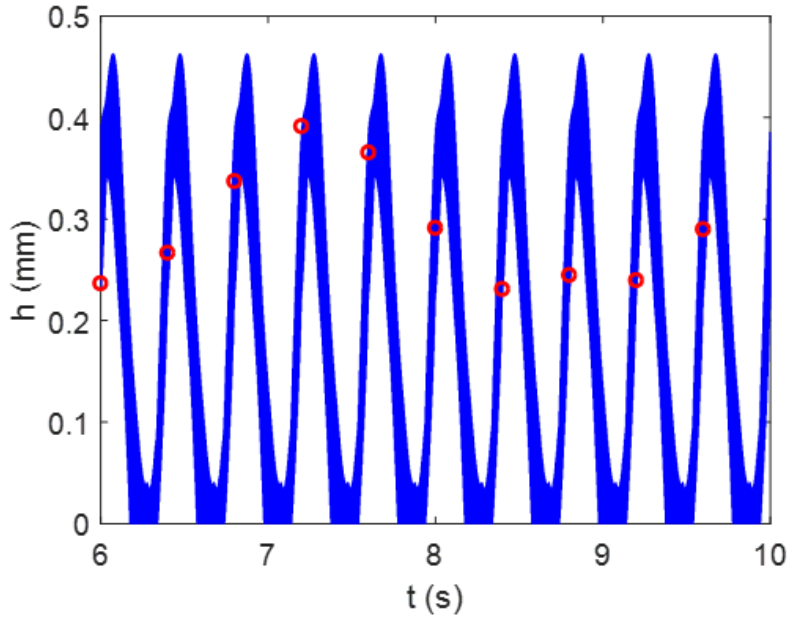

Figure 3.12: Time domain chip thickness for RAF = 1.0 and OPR = 0.25.

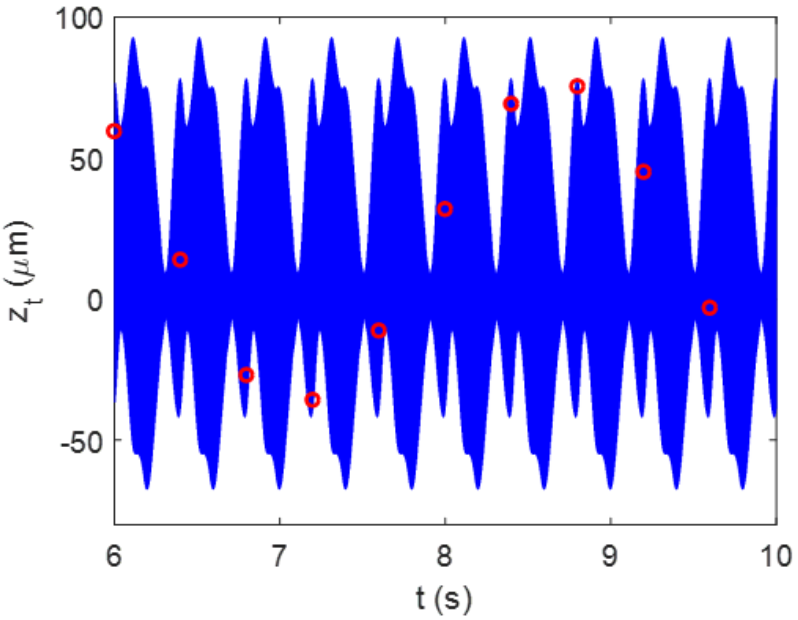

Figure 3.13: Time domain tool displacement for RAF = 1.0 and OPR = 0.25.

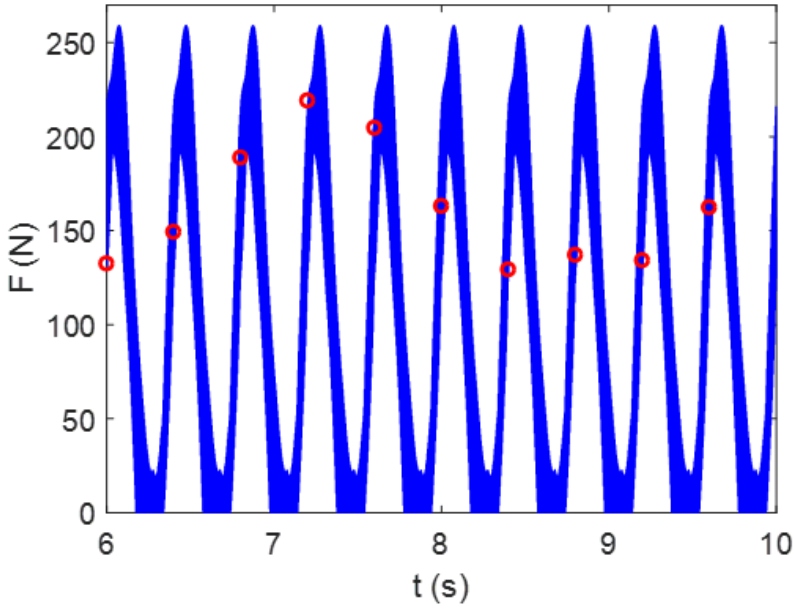

Figure 3.14: Time domain resultant force for RAF = 1.0 and OPR = 0.25.

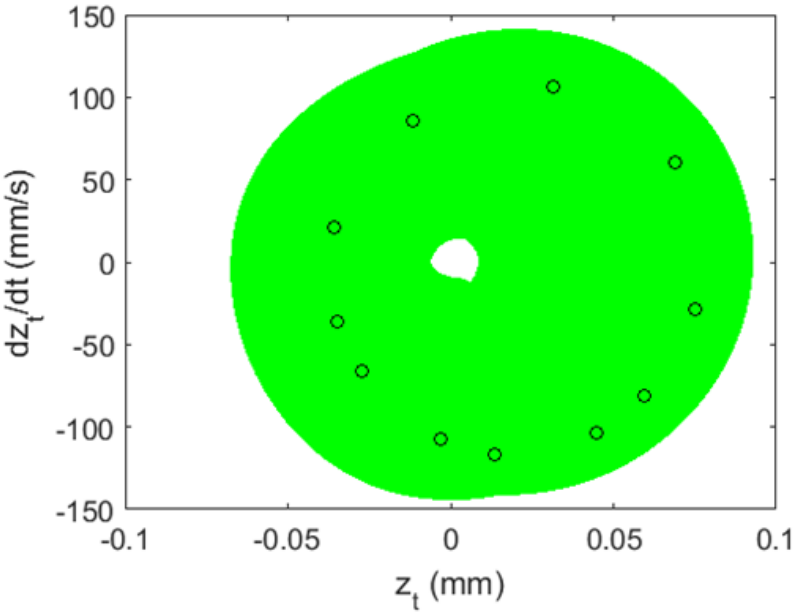

Figure 3.15: Poincaré map of sampled tool displacement versus velocity for RAF = 1.0 and OPR = 0.25.

CHAPTER FOUR
MATERIALS AND METHODS: MTP SURFACE FINISH

Surface finish simulation

In order to model the workpiece surface finish, the previously described time domain simulation is modified to predict the surface profile left on a workpiece. The simulation modification details are provided in the following paragraphs.

Include 3D dynamics

The initial portion of the time domain simulation remains the same when calculating the tool motion (Eq. 2) and instantaneous chip thickness (Eq. 5). The first modification that is made to the time domain simulation is to include the cutting tool's radial direction structural dynamics. The orientation of the radial direction, u_3, is shown in Figure 4.1. Once the instantaneous uncut chip thickness, h, for the time step is calculated, the radial direction force, F_r, is computed using Eq. 19

$$F_r = k_r b h \tag{19}$$

where k_r is the radial cutting force coefficient and b is the depth of cut (chip width).

As shown in Eqs. 9 and 10, a power law relationship enables the effect of the instantaneous chip thickness to be incorporated in cutting force coefficients. The revised radial force model is defined in Eq. 20

$$F_r = (a_r h^{b_r} + c_r) b h \tag{20}$$

where a, b, and c denote power law fitting values and the r subscript denotes the radial direction.

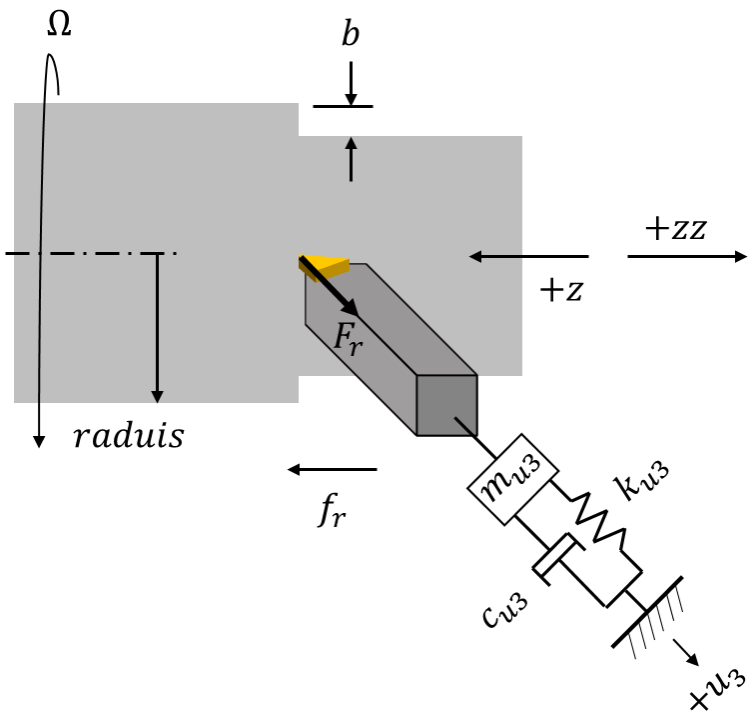

Figure 4.1: Flexible tool MTP turning surface finish model. The radial force, F_r, component is identified as well as the modal parameters that represent the structural dynamics in the orthogonal radial direction u_3. The MTP feed motion, z, and total motion, zz, are also identified.

The radial force is then projected into the u_3 mode direction. Note that since the radial force is the only force in the u_3 direction, the F_{u3} force is simply calculated by Eq. 21.

$$F_{u3} = F_r = k_r bh \tag{21}$$

The Euler integration procedure is the same as described previously to solve for the displacements in the u_1 and u_2 directions in Eqs. 13 and 14. The procedure is updated with the force, modal mass, damping, and stiffness values for the u_3 direction and shown in Eq. 22. Once the u_3 displacement is known, it is stored in a vector to be later referenced to simulate the workpiece surface profile.

$$\begin{aligned} \ddot{u}_3 &= \frac{F_{u3} - c_{u3}\dot{u}_3 - k_{u3}u_3}{m_{u3}} \\ \dot{u}_3 &= \dot{u}_3 + \ddot{u}_3 dt \\ u_3 &= u_3 + \dot{u}_3 dt \end{aligned} \tag{22}$$

Include surface finish

The second modification to the time domain simulation is to add the nose radius geometry to predict the profile that is imparted on the workpiece surface. Previously, the MTP tool position was parsed by the workpiece revolution and plotted (see Figure 3.2, Figure 3.3, and Figure 3.4). Figure 4.2 displays the MTP feed motion for a spindle speed of 200 rpm, a feed of 0.1 mm/rev, and OPR and RAF values of 0.5 and 0.8, respectively. Figure 4.2 is similar to Figure 3.3 and Figure 3.4. However, where Figure 3.3 and Figure 3.4 plotted against time, in

Figure 4.2 the MTP tool motion is plotted against the surface distance, *d*, traveled in a single revolution (the workpiece circumference).

In Figure 4.2, the solid line denotes the MTP feed advance of the tool into the part. The revolution numbers are included on the right hand of the figure to aid the reader in distinguishing the individual revolutions. The user then selects some predefined distance along the workpiece circumference to set the nose radius test location. The MTP tool position is parsed at the user specified nose radius test location for each revolution and stored in a vector, z_{parse}. In Figure 4.2 a nose radius test location halfway through the workpiece circumference was selected and is denoted by red circles.

This process is repeated with the simulated radial tool displacement by parsing each revolution and the user specified nose radius test location. The resulting radial tool displacement values are resaved in a vector, n_{parse}. The two vectors that were created from parsing (z_{parse} and n_{parse}) are the position along the feed direction from the end of the part and radial tool position locations, respectively. Now that the tool locations are known, the shape of the nose radius of the tool can be modeled. The lower half of the circle equation is used to model the tool's nose radius, see Eq. 23.

$$n = -(n_r^2 - (z - z_o)^2)^{\frac{1}{2}} + n_o \tag{23}$$

In Eq. 23, n is the radial position of the tool nose radius along the radial direction, n_r is the specified nose radius of the tool, z is the axial position along the feed direction.

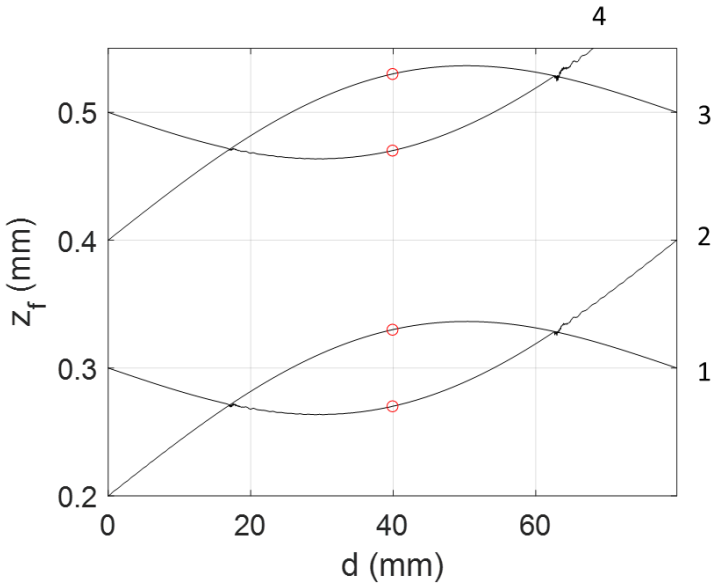

Figure 4.2: MTP tool feed position for four spindle revolutions. The spindle speed is 200 rpm, the feed per revolution is 0.1 mm/rev, and the OPR and RAF values are 0.5 and 0.8. The user selected nose radius test location is shown using circles.

Also, z_o and n_o are the center location of the circle in the feed direction from the end of the workpiece and the radial position, respectively, for a single revolution. The tool's nose radius position and shape are calculated for each revolution and overlaid on top of the previous revolution's nose radius. Using Eq. 23 and the example parameters from Figure 4.2, four revolutions of the nose radii are superimposed on top of each other in Figure 4.3. In the figure, the solid line describes the outer edge of the tool nose radius for each revolution. Numbers that correspond to the revolutions in Figure 4.2 are provided to aid the reader in identifying the four nose radii position and shape from each revolution.

To extract the predicted surface profile that is transferred from the tool's nose radius to the workpiece, only the minimum portion of the overlaid tool nose radii is selected. The lower portion of the overlaid tool nose radii is the final surface profile that is left on the workpiece. Using the same example parameters form Figure 4.2, the final predicted surface profile is displayed in Figure 4.4.

Roughness calculation

Using the predicted workpiece surface profile shown in Figure 4.4, the arithmetic mean surface roughness, R_a, of the surface profile is calculated using Eq. 24

$$R_a = (|x_1| + |x_2| + |x_3| + \cdots + |x_N|)/N \tag{24}$$

where x is the surface profile height value and N is the total number of sampled points that make up the surface profile.

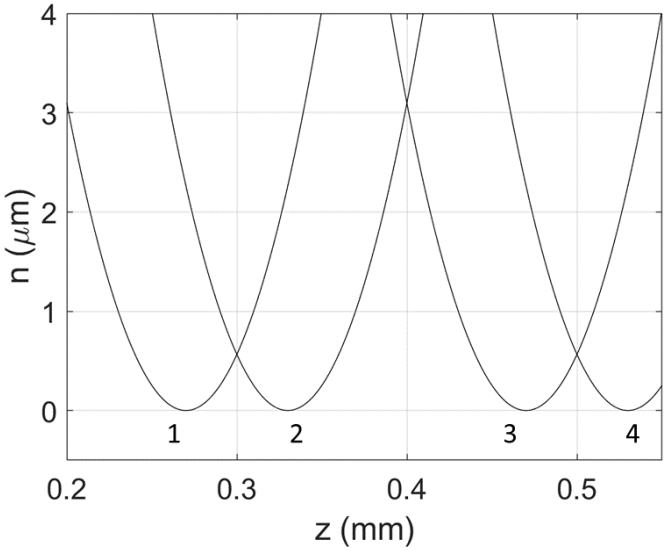

Figure 4.3: Tool nose radius geometry overlay plots. The plot shows the sampled nose radii from revolutions 1-4 from Figure 4.2.

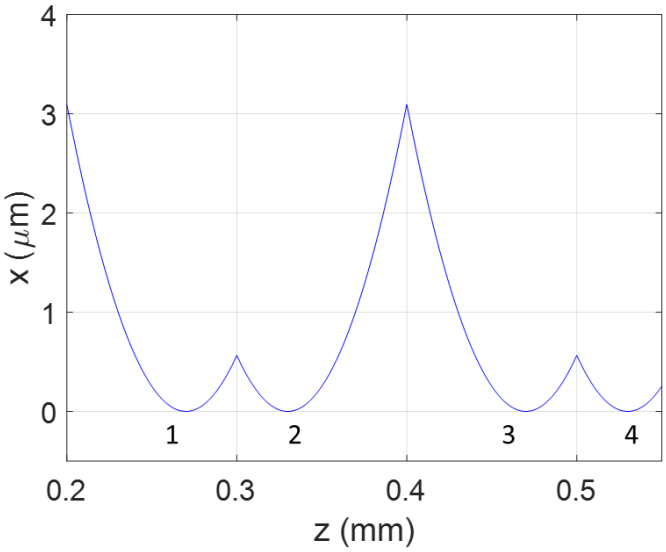

Figure 4.4: Predicted workpiece surface profile for revolutions 1-4 from Figure 4.2.

Selected locations on the MTP parameter map (see Figure 1.5) can now be investigated further with additional measured and predicted surface profile comparisons. To include the effects of the actual chip formation on the surface profile, previous work in modeling the effect of plastic side flow around the tool edge on surface finish by Kai and Melkote [39] has been included. Since the MTP chip thickness varies from zero to some peak value (see Figure 3.5) it is necessary to model the effective tool nose radius to account for effects of side flow at low chip thicknesses.

To facilitate a side flow model, Eq. 22 (circle equation that describes the tool nose radius) is updated to the form in Eq. 25

$$n = -((ah^b + c)^2 - (z - z_o)^2)^{\frac{1}{2}} + n_o \tag{25}$$

where a, b, and c are power law fitting coefficients that describe the effective nose radius. The relationship between the effective nose radius and commanded chip thickness is obtained by completing constant feed turning tests and fitting the nose radius to match the measured surface profile over a range of chip thickness values.

Summary: Modifications were made to the previously described time domain simulation to account for the radial tool dynamics and to predict the generated workpiece surface profile. A revised effective nose radius model was provided to account for plastic side flow around the tool's nose radius.

CHAPTER FIVE
EXPERIMENTAL SETUP

MTP turning stability

The testbed for the turning experiments was a Haas TL-1 CNC lathe (8.9 kW maximum spindle power, 2000 rpm maximum spindle speed). Tubular workpieces were machined from AISI 1026 drawn over mandrel steel. To keep a consistent surface speed across multiple workpieces, the tubular workpieces were machined to have a mean diameter of 70 mm. The wall thicknesses were 3.5 mm and 4.5 mm. Concentricity and cylindricity of the outside and inside diameters with the rotational axis of the lathe spindle was assured by indicating the workpiece into alignment prior to conducting the experiments. Type C, 80° parallelogram carbide inserts with a zero-rake angle, 7° relief angle, and a flat rake face were used (ANSI catalog number CCMW3252, Kennametal part number 3757916). A flexure-based cutting tool cutting tool was manufactured such that the dynamics of the cutting tool exhibit stable and unstable cutting for the available machining setup and machining parameters. Tube turning was selected so that orthogonal cutting could be approximated. All experiments were conducted at a mean cutting speed of 122 m/min (556 rpm) with a nominal feed rate of 0.102 mm/rev. Stability of the cuts was controlled by varying the tube wall thickness (i.e., the chip width) for various RAF and OPR pairs.

Instruments that were included to facilitate in-process metrology of the cutting tests include: 1) a three-axis dynamometer (Kistler 9257B) mounted to the

cross slide to measure the dynamic cutting forces; 2) a laser vibrometer (Polytec OFV-534/OFV-5000) was used to measure the feed direction, z_f, velocity of the cutting tool and a capacitance probe (Lion Precision C-18-13-2.0) were used to measure tool displacement, z_t; and 3) a laser tachometer was used to determine the actual spindle speed for periodic sampling at the MTP forcing frequency. See Figure 5.1, where the normal direction was aligned with the spindle axis, while the tangential direction was tangent to the cut surface (vertical).

The tool's frequency response function was measured using impact testing [36] . The results are displayed in Figure 5.2 and Figure 5.3. Modal fitting was completed to extract the modal mass, m, viscous damping, c, and stiffness, k, values for the time domain simulation [36] . The flexure-based cutting tool's modal parameters are reported in Table 5.1.

The coefficients for the cutting force model were identified from continuous (stable) cutting tests using the selected work material and insert. The cutting force components in the normal and tangential directions were measured by the dynamometer for known chip thickness and width values. This process was carried out for decreasing chip thickness values until the cutting test no longer forced a continuous chip. This was done to properly model the effect on cutting force coefficients as chip thickness decreases, which is essential to model due to MTP being an interrupted cutting process. The coefficients were then extracted using Eqs. 7-8. The known chip thickness and width value combinations were then fitted with a power law shown in Eqs. 26-27 and Figure 5.4.

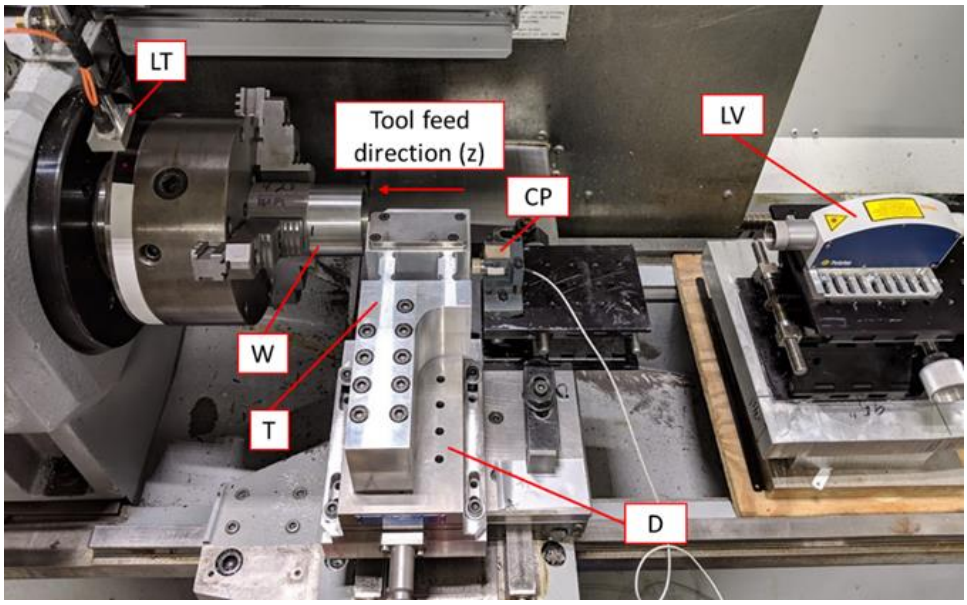

Figure 5.1: Photograph of tube turning setup including workpiece (W), dynamometer (D), flexure-based cutting tool (T), laser tachometer (LT), laser vibrometer (LV), and capacitance probe (CP).

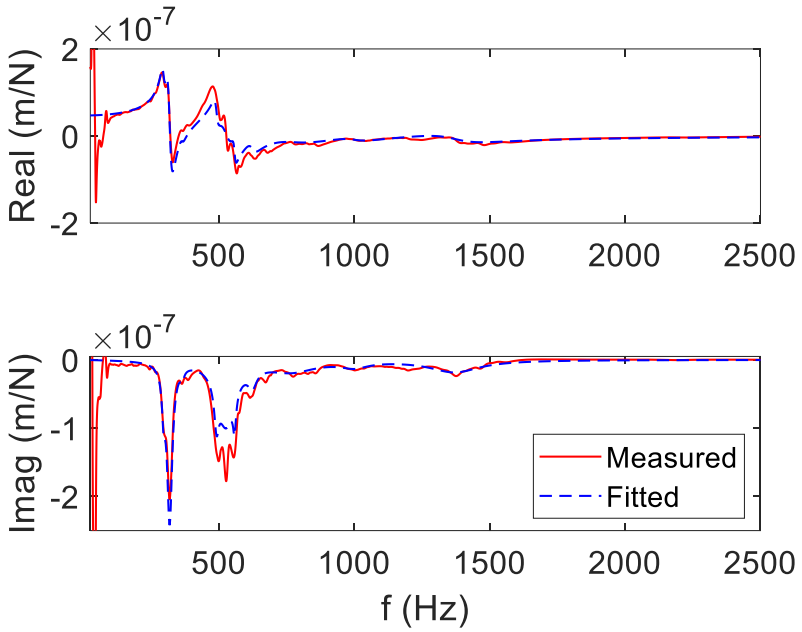

Figure 5.2: Stability testing tool point frequency response function for the normal (feed) direction. (Top) real part; and (bottom) imaginary part.

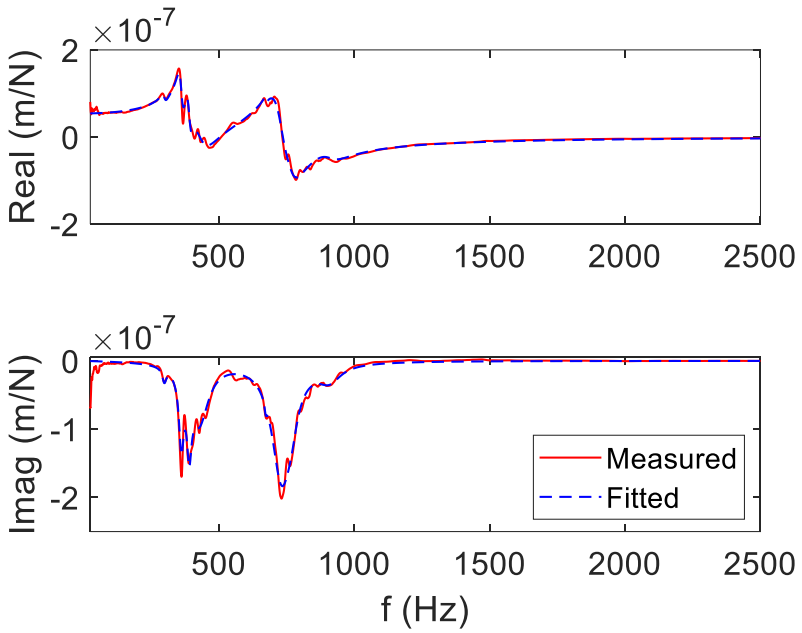

Figure 5.3: Stability testing tool point frequency response function for the tangential (tool height) direction. (Top) real part; and (bottom) imaginary part.

Table 5.1. Modal parameters of the flexure-based cutting tool (T) in Figure 5.1

Normal direction

Mode	m (kg)	c (N-s/m)	k (N/m)
1	99.2	9.07×10^3	3.43×10^8
2	14.8	2.16×10^3	5.9×10^7
3	330	3.31×10^4	1.73×10^9
4	45.9	5.66×10^3	4.37×10^8
5	6.82	3.38×10^3	7.50×10^7
6	69.6	5.86×10^3	8.55×10^8
7	42.3	1.03×10^4	6.49×10^8
8	9.57	1.32×10^4	2.27×10^8
9	24.4	1.90×10^4	9.69×10^8
10	4.63	6.60×10^3	3.46×10^8

Tangential direction

Mode	m (kg)	c (N-s/m)	k (N/m)
1	252	2.77×10^4	8.85×10^8
2	33.6	4.44×10^3	1.73×10^8
3	33.9	4.44×10^3	2.04×10^8
4	8.22	4.36×10^3	5.97×10^7
5	295	1.75×10^4	5.27×10^9
6	2.43	1.34×10^3	5.15×10^7
7	13.3	5.13×10^3	3.09×10^8
8	11.4	7.46×10^3	3.73×10^8

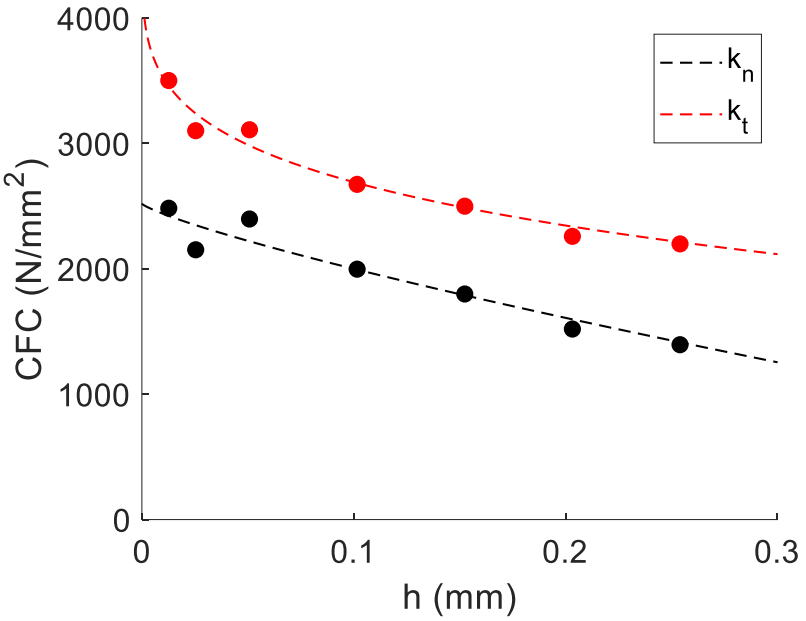

Figure 5.4: Normal and tangential direction cutting force coefficients (CFC) fits (dashed lines) with individual continuous cutting tests points (dots).

Since MTP is an interrupted cutting operation, the MTP cutting force coefficient model is analogous to a milling cutting force coefficient model in that it exhibits an edge coefficient [38] .

$$k_n = -3355 * h^{0.81} + 2520, \quad (26)$$

$$k_t = -3490 * h^{0.22} + 4795, \quad (27)$$

Summary: An in-process metrology orthogonal (tube) turning setup was described. Modal parameters were measured and reported for the flexure-based cutting tool. The modeled and fitted frequency response functions were displayed. A power law cutting force model was used to capture the effect of the cutting force at small chip thickness values; it was provided and plotted.

MTP surface finish

To facilitate comparing measured and predicted surface profiles for outer diameter finish turning tests, a Haas TL-1 CNC lathe (8.9 kW maximum spindle power, 2000 rpm maximum spindle speed) was utilized. The proposed surface finish tests were conducted using 6061-T6 aluminum workpieces with a 0.127 mm commanded depth of cut, a 0.051 mm/rev commanded global feed rate, and various MTP parameters (OPR and RAF pairs). A VMBT-331 (35-degree diamond with a 0.397 mm nose radius) carbide insert was selected. A new cutting insert of the same geometry was used for each cutting test to eliminate the effects of varying tool nose radius due to tool wear and/or material weld back to the insert.

To measure the actual feed motion (both with and without MTP) a Keyence LK-H157 laser triangulation displacement sensor was used. The laser triangulation

displacement sensor was mounted to the machine tool's Z-axis way using a magnetic base mount. A photograph of the testing setup is shown in Figure 5.5.

The tool and workpiece frequency response functions were measured using impact testing. Modal fitting was then employed to extract the modal mass, *m*, viscous damping, *c*, and stiffness, *k*, values for the simulation. The resulting modal fitting parameters are presented in Table 5.2 - Table 5.4. The comparisons between the tool and workpiece measured and fitted frequency response functions are provided in Figure 5.6 - Figure 5.8.

A Mitutoyo Contracer was used to provide post-process metrology to capture and record the test workpiece surface profiles. The test workpiece was placed on a positioning stage such that the feed direction was aligned with the measurement direction of the profilometer. The Mitutoyo Contracer consists of a profilometer stylus that is placed on the test workpiece surface. The stylus is then moved across the test surface and the surface deviations are detected by a controller unit that is mounted to a granite surface plate. The test workpiece is placed on a workpiece positioning stage that sits on top of the granite surface plate. The surface profile data is transmitted via a USB connection to a computer with a Mitutoyo software application that reads and saves the measured surface profile for further data analysis. A schematic of the surface profilometry setup used to measure the test workpiece surface profiles is provided in Figure 5.9. Figure 5.10 provides a cartoon diagram of the surface profile trace in relation to the machining feed direction and the machine tool's coordinate system. The measured surface profile trace is denoted by the red line on the example test workpiece.

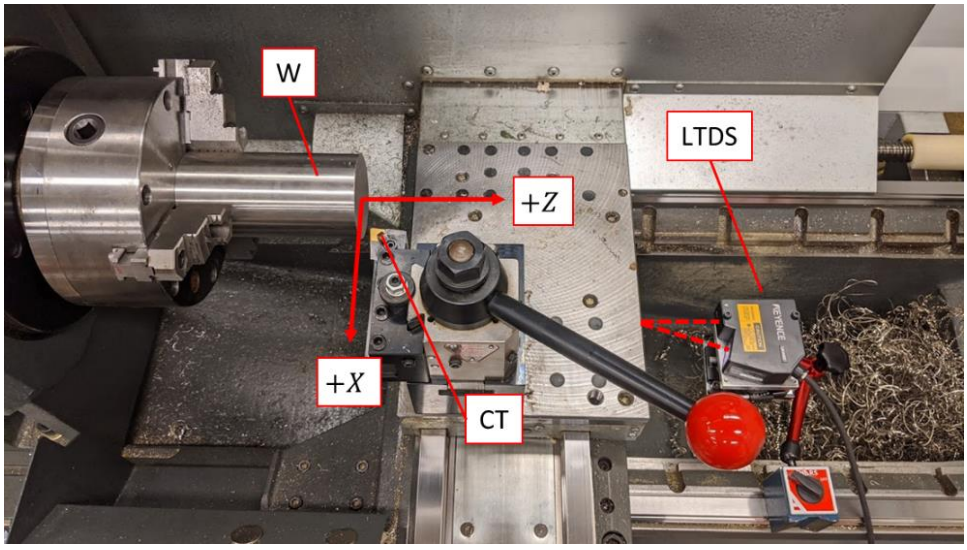

Figure 5.5: Photograph of outer diameter turning setup including workpiece (W), cutting tool (CT), and laser triangulation displacement sensor (LTDS).

Table 5.2. Modal parameters of the cutting tool (T) and workpiece (W) in the normal (feed) direction for the setup shown in Figure 5.5. The tabulated vales correspond to the frequency response function shown in Figure 5.6.

Tool dynamics

Mode	m (kg)	c (N-s/m)	k (N/m)
1	31.2	2.79×10^4	1.75×10^8
2	0.5	9.75×10^4	1.37×10^8
3	0.1	1.15×10^5	5.79×10^7
4	0.1	1.54×10^5	6.05×10^7
5	0.03	7.40×10^4	3.87×10^7
6	0.1	7.09×10^4	1.12×10^8

Workpiece dynamics

Mode	m (kg)	c (N-s/m)	k (N/m)
1	189.2	4.73×10^4	3.17×10^8
2	1776.3	1.33×10^5	4.56×10^9
3	308.1	1.11×10^5	1.03×10^9
4	106.6	6.17×10^4	4.15×10^8
5	693.2	8.92×10^4	4.27×10^9
6	10.5	1.59×10^5	1.27×10^9
7	12.2	1.40×10^5	3.73×10^9
8	3.8	1.83×10^5	1.65×10^9
9	7.7	5.13×10^5	6.27×10^9

Table 5.3. Modal parameters of the cutting tool (T) and workpiece (W) in the tangential (tool height) direction for the setup shown in Figure 5.5. The tabulated vales correspond to the frequency response function shown in Figure 5.7.

Tool dynamics

Mode	m (kg)	c (N-s/m)	k (N/m)
1	0.1	2.97×10^{5}	6.77×10^{7}
2	0.2	2.42×10^{5}	1.53×10^{8}
3	0.1	1.23×10^{5}	6.38×10^{7}
4	0.1	8.93×10^{4}	1.24×10^{8}
5	0.04	8.19×10^{4}	6.13×10^{7}
6	0.1	1.79×10^{5}	1.71×10^{8}

Workpiece dynamics

Mode	m (kg)	c (N-s/m)	k (N/m)
1	86.3	3.58×10^{4}	1.47×10^{8}
2	8.8	1.61×10^{4}	3.43×10^{7}
3	88.1	6.47×10^{4}	4.61×10^{8}
4	35.9	3.11×10^{4}	2.23×10^{8}
5	21.1	1.42×10^{5}	2.74×10^{8}
6	4.8	6.42×10^{4}	1.42×10^{8}
7	44.7	4.46×10^{4}	1.39×10^{9}
8	5.5	2.18×10^{5}	2.16×10^{8}
9	2.6	4.58×10^{4}	1.80×10^{8}
10	2.9	1.90×10^{4}	2.11×10^{8}

Table 5.4. Modal parameters of the cutting tool (T) and workpiece (W) in the radial direction for the setup shown in Figure 5.5. The tabulated vales correspond to the frequency response function shown in Figure 5.8.

Tool dynamics			
Mode	m (kg)	c (N-s/m)	k (N/m)
1	5678.2	3.77×10^4	8.07×10^8
2	4814.2	9.26×10^4	2.47×10^9
3	85.6	2.06×10^4	7.20×10^7
4	344.8	1.22×10^5	4.46×10^8
5	66.9	7.19×10^4	1.87×10^8
6	442.3	1.61×10^5	1.89×10^9
7	25.2	1.48×10^5	2.67×10^8
8	113.9	1.62×10^5	1.40×10^9
9	52.0	2.53×10^5	2.62×10^9
10	2.2	2.60×10^5	6.35×10^8

Workpiece dynamics			
Mode	m (kg)	c (N-s/m)	k (N/m)
1	17.5	1.52×10^4	6.77×10^7
2	59.2	5.03×10^4	2.96×10^8
3	35.8	7.55×10^4	2.75×10^8
4	24.5	3.17×10^4	3.53×10^8
5	37.0	4.44×10^4	5.99×10^8
6	25.1	1.00×10^5	6.04×10^8
7	3.1	8.34×10^4	1.16×10^8
8	1.9	4.65×10^4	1.35×10^8
9	3.8	3.15×10^4	2.82×10^8

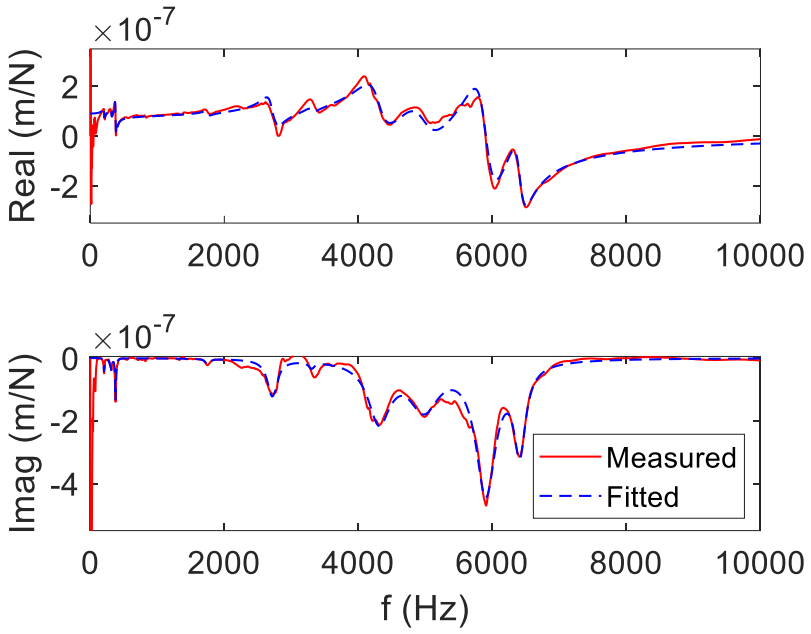

Figure 5.6: Surface finish tool point and workpiece free end frequency response function for the normal (feed) direction. (Top) real part; and (bottom) imaginary part.

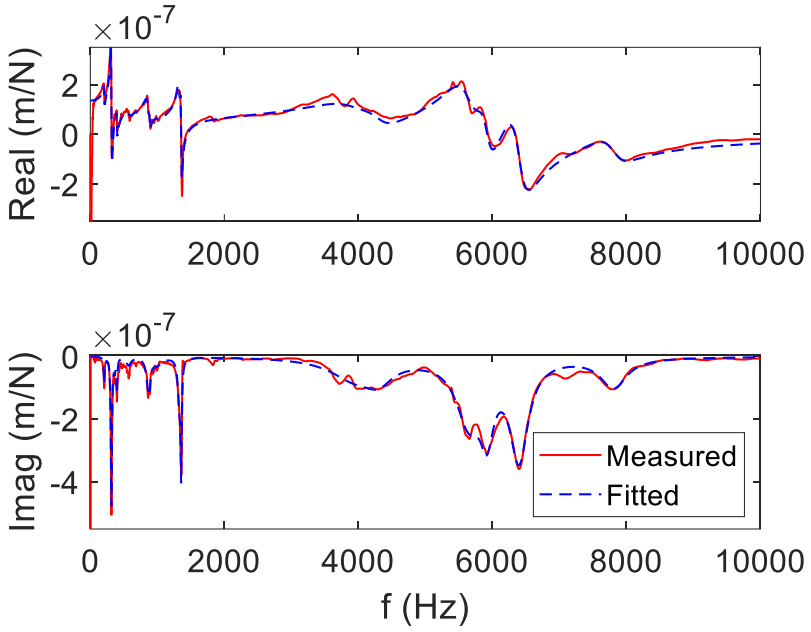

Figure 5.7: Surface finish tool point and workpiece free end frequency response function for the tangential (tool height) direction. (Top) real part; and (bottom) imaginary part.

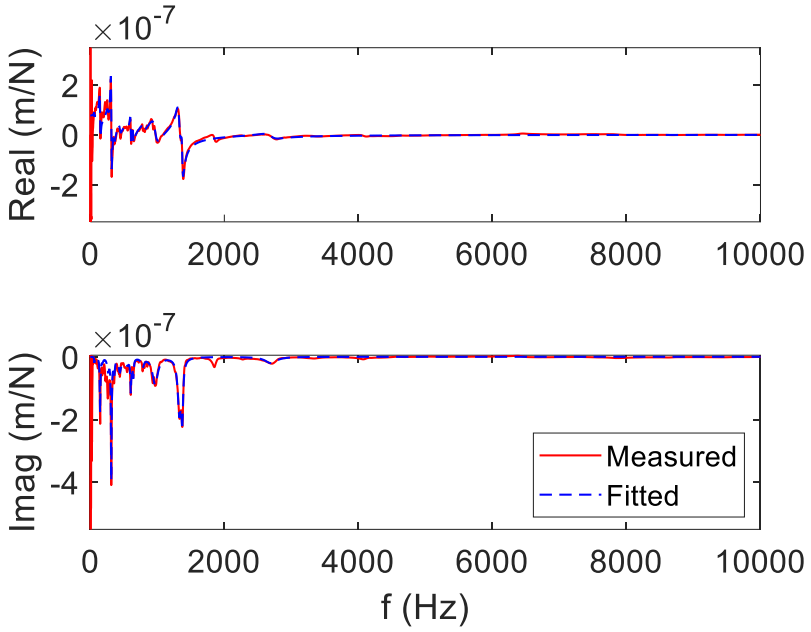

Figure 5.8: Surface finish tool point and workpiece free end frequency response function for the radial direction. (Top) real part; and (bottom) imaginary part.

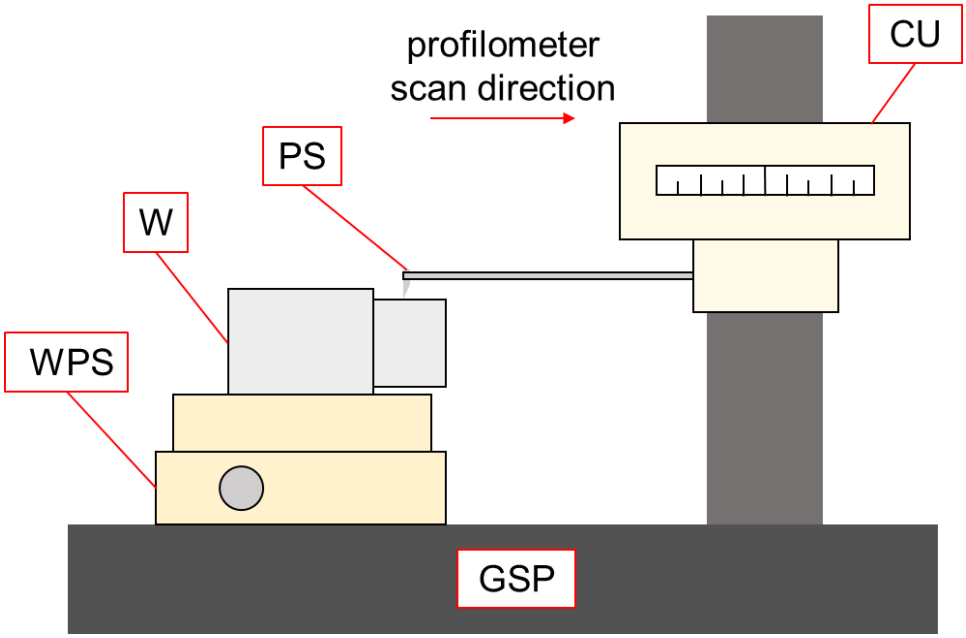

Figure 5.9: Schematic of the Mitutoyo Contracer surface profilometry setup including granite surface plate (GSP), workpiece (W), controller unit (CU), and profilometer stylus (PS) and workpiece positioning stage (WPS).

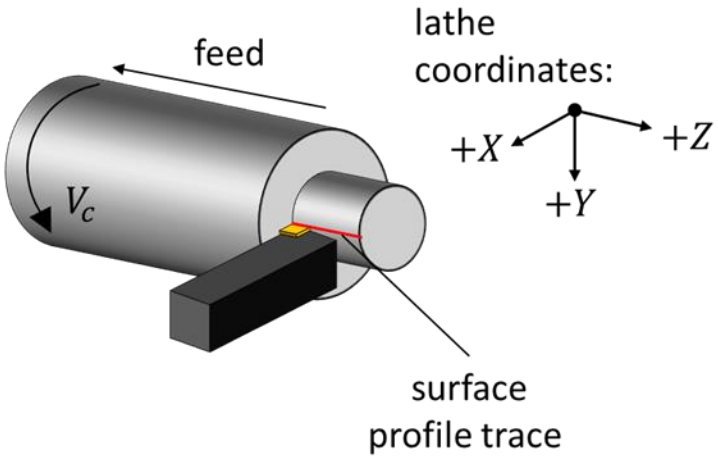

Figure 5.10: Schematic of the surface profile locations and direction relative to the machining feed direction and machine tool's coordinate system. The surface profile trace is denoted by the red line on the workpiece.

As previously described, the power law cutting force model was identified experimentally. The only difference is the inclusion of Eq. 19 when extracting the coefficients to account for the radial direction force. The known chip thickness and width value combinations were then fitted with a power law shown in Eqs. 28-30. The individual continuous cutting tests points for both the normal, tangential, and radial directions with their corresponding power law fits are shown in Figure 5.11.

$$k_n = 39.1 * h^{-0.97} + 111.8, \quad (28)$$

$$k_t = 131.1 * h^{-0.89} + 702.1, \quad (29)$$

$$k_r = 124.1 * h^{-0.99} + 564.4, \quad (30)$$

Summary: A setup to conduct finish turning surface finish tests with in-process metrology to measure the actual tool path motions was detailed. The structural dynamics modal parameters were tabulated and plotted for the selected cutting tool and workpiece. A post-process metrology setup that provides surface profilometry measurements was also shown. A cutting force model that was developed for the selected workpiece and insert was provided and plotted.

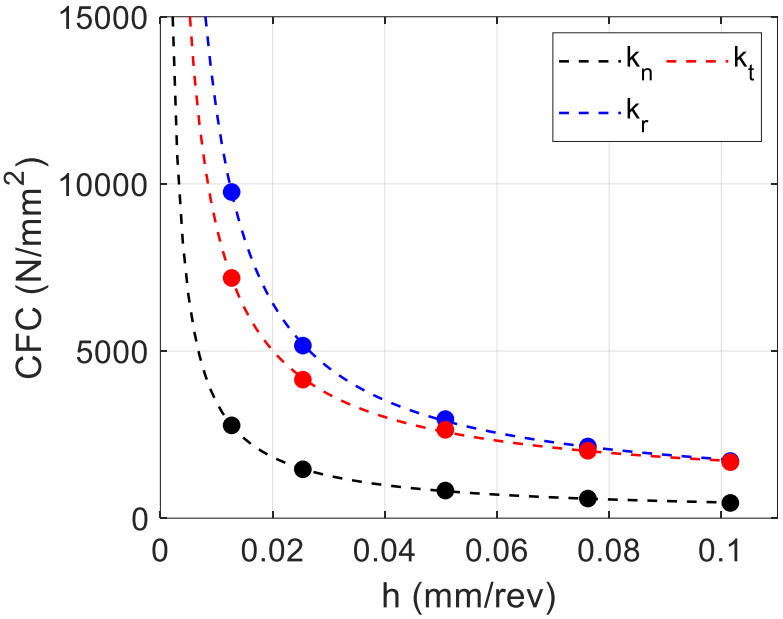

Figure 5.11: Normal, tangential, and radial direction cutting force coefficients (CFC) fits (dashed lines) with individual continuous cutting tests points (dots).

CHAPTER SIX
RESULTS AND DISCUSSION

MTP turning stability

Time domain simulations were completed on grids of $\{RAF, OPR\}$ pairs from 0 to 3 in steps of 0.05 for individual chip width (tube wall thickness) values, where the modal parameters, force model, spindle speed, and nominal feed from the previous section were applied. The M value was computed for each $\{RAF, OPR\}$ pair and recorded. A stability map was then produced for each discrete chip width value by plotting a single contour at M = 2 µm. This contour separated stable $\{RAF, OPR\}$ combinations from unstable combinations, i.e., M > 2 µm points were considered unstable and M ≤ 2 µm were considered stable. Additionally, the analytical chip breaking limit that identifies nominal $\{RAF, OPR\}$ combinations which provide discontinuous chips was also superimposed on each stability map [44].

Time domain simulation generated stability maps

An example stability map is provided in Figure 6.1, where the chip width is 3.5 mm. In this case, stable conditions are observed for all $\{RAF, OPR\}$ combinations so the entire map is white. The dashed lines identify the analytical chip breaking limit. Only the areas enclosed by the lines provide discontinuous chips. For example, $\{RAF, OPR\}$ = {1, 0.5} theoretically breaks chips, while $\{RAF, OPR\}$ = {1, 1} does not. For $\{RAF, OPR\}$ = {1, 0.5}, the tool path is repeated every two revolutions causing the tool to exit the cut and break the chip. Whereas $RAF, OPR\}$ = {1, 1} the cutting tool repeats the same tool motion every revolution.

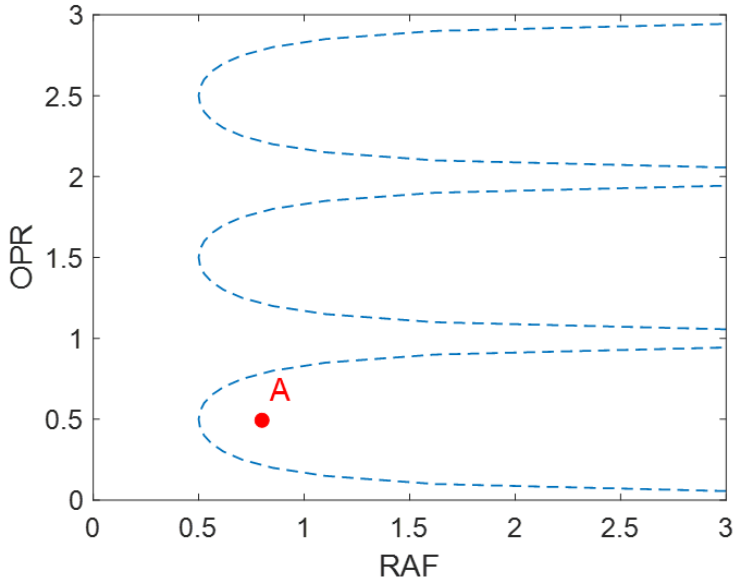

Figure 6.1: Stability map for b = 3.5 mm. All {RAF, OPR} pairs are stable for this chip width, so the background is all white. The analytical chip breaking limit is identified by the dashed lines. Individual test points are denoted by red circles and a letter.

A second stability map is displayed in Figure 6.2; the chip width is 4.5 mm. With the increased chip width, some $\{RAF, OPR\}$ combinations are stable, and others are not. It is interesting to observe that the stability limit approximately follows the analytical chip breaking limit. This indicates that, even though all points (except {0, 0}) in the map exhibit oscillating tool motion, the stability is increased when chip breaking actually occurs and the force drops instantaneously to zero. This highlights the difference between MTP and continuous cutting. MTP has an inherent forcing frequency and is, therefore, a fundamentally different process than continuous turning. It is a hybrid between milling, where the time-dependent chip thickness is defined by the trochoidal teeth trajectories for the rotating and translating endmill, and turning, where the chip thickness is ideally constant and set by the feed per revolution.

A third stability map is displayed in Figure 6.3, where the chip width has been increased to 5 mm. The stable region is now smaller with increasing stability at higher OPR values. The higher oscillating frequency, which is the product of the spindle speed and OPR ($OPR = \frac{60*f}{\Omega}$), tends to sustain the stable behavior. Note that the chip width can be increased to a level where the entire map is unstable.

Cutting test parameter locations

To verify the stability predictions, cutting tests were performed at various $\{RAF, OPR\}$ combinations for different chip widths. The selected cutting test location parameters, denoted in Figure 6.1 and Figure 6.2 by red dots and letters A-E, are tabulated in Table 6.1, respectively.

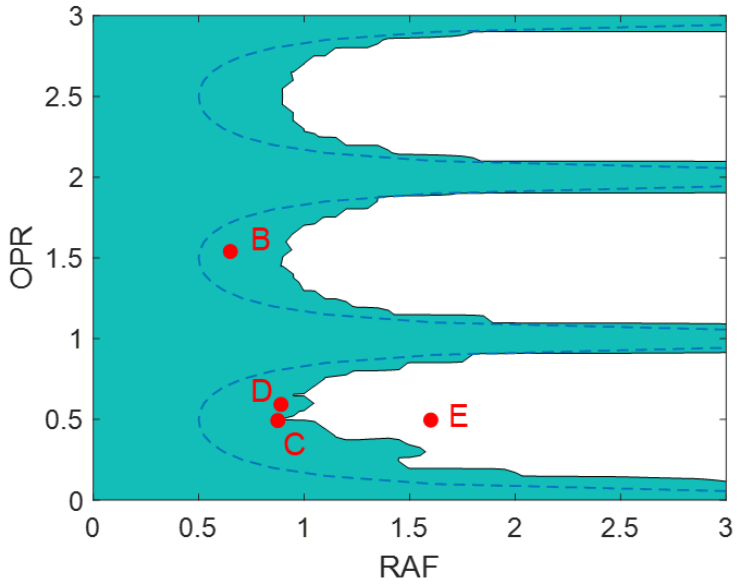

Figure 6.2: Stability map for b = 4.5 mm. Only selected {RAF, OPR} pairs are stable for this chip width; stable combinations are identified by the white background, while the dark background indicates unstable behavior. Individual test points are denoted by red circles and a letter.

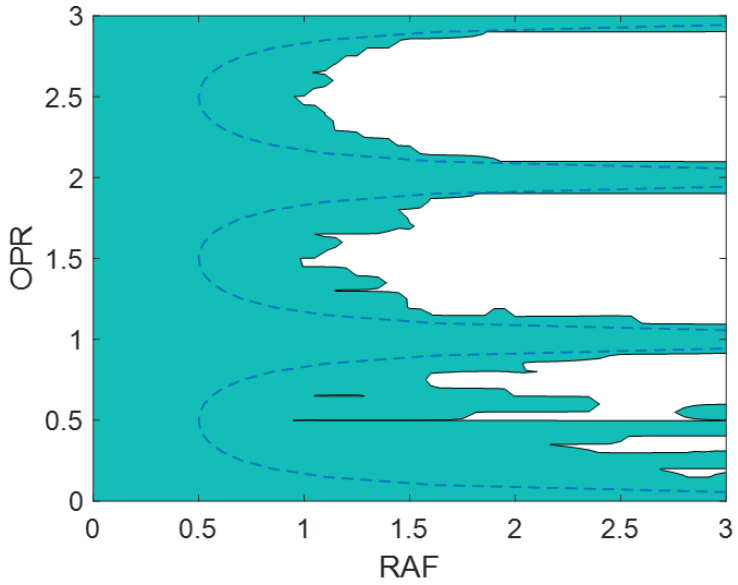

Figure 6.3: Stability map for b = 5 mm. The stability zone size is reduced relative to Figure 6.2 with a smaller chip width.

Table 6.1. Cutting stability tests naming convention. The desired, commanded, and measured $\{RAF, OPR\}$ pairs are provided for each cutting test.

		Desired		Commanded		Measured	
Point	b (mm)	RAF	OPR	RAF	OPR	RAF	OPR
A	3.5	0.8	0.5	2	0.5	0.8	0.494
B	4.5	0.6	1.55	18	1.6	0.64	1.54
C	4.5	0.8	0.5	2	0.5	0.875	0.492
D	4.5	0.8	0.6	3.25	0.6	0.89	0.593
E	4.5	1.5	0.5	3.25	0.5	1.6	0.496

For each test location the chip width in mm, RAF, and OPR values are provided for the desired, commanded, and measured RAF and OPR pairs. The actual tool displacement was measured in-situ using the laser vibrometer shown in Figure 5.1 and fitted post process. Examples of the modeled and measured tool displacement are provided in Figure 6.4 and Figure 6.5 for test points A and point B respectively. Note in the measured tool displacement profile in Figure 6.5 an high frequency content is present. This is due to the presence of unstable cutting behavior that presents itself as the chatter frequency.

Cutting test results

The measured and predicted time domain cutting force and tool displacements were compared, as well as the periodic sampling results. Examples are provided in Figure 6.6 - Figure 6.10 and Figure 6.11 - Figure 6.16. In Figure 6.6 - Figure 6.10, the normal direction cutting force, tool displacement, and Poincaré map are provided for test point A (b = 3.5 mm, RAF = 0.8, OPR = 0.494). Figure 6.6 shows the predicted and measured cutting force in the normal direction. The periodic sampling points are shown by the dark circles and are overlaid on the force signals. Good agreement is observed between the predicted and measured cutting force signals with similar profiles and magnitudes. The predicted and measured tool displacement, shown in Figure 6.7, is a similar result with matching displacement profiles and magnitudes. Figure 6.8 shows the predicted and measured Poincaré map. As seen in the normal cutting force and tool displacement, the predicted and measured signals match each other.

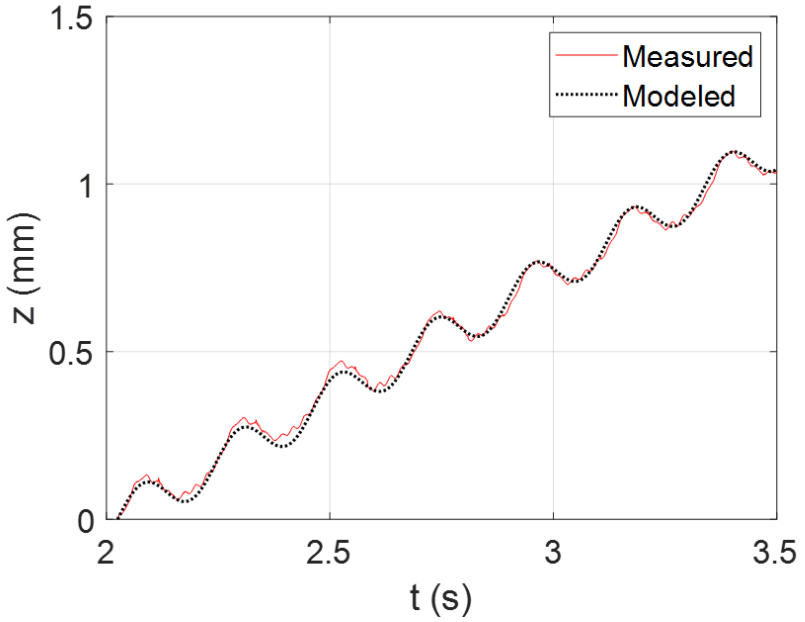

Figure 6.4: Modeled and measured tool displacement in feed direction test point A.

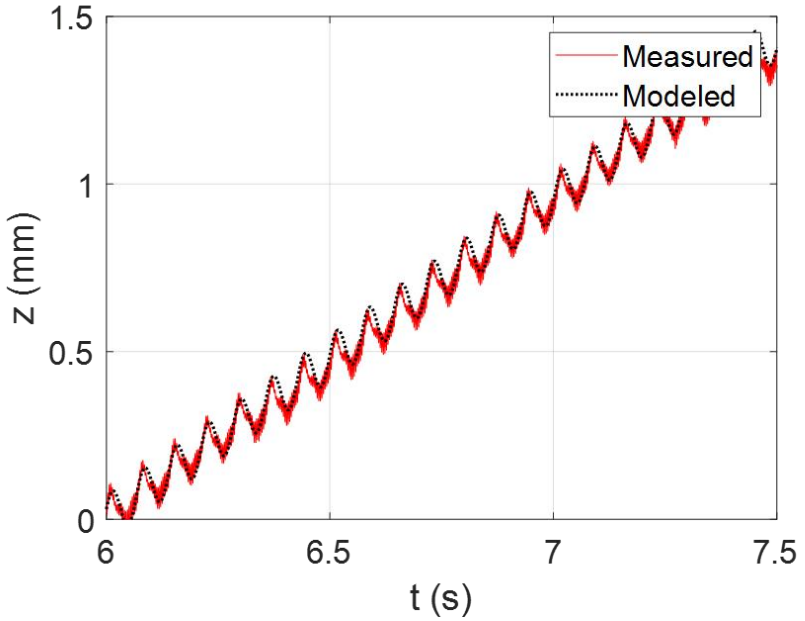

Figure 6.5: Modeled and measured tool displacement in feed direction test point B.

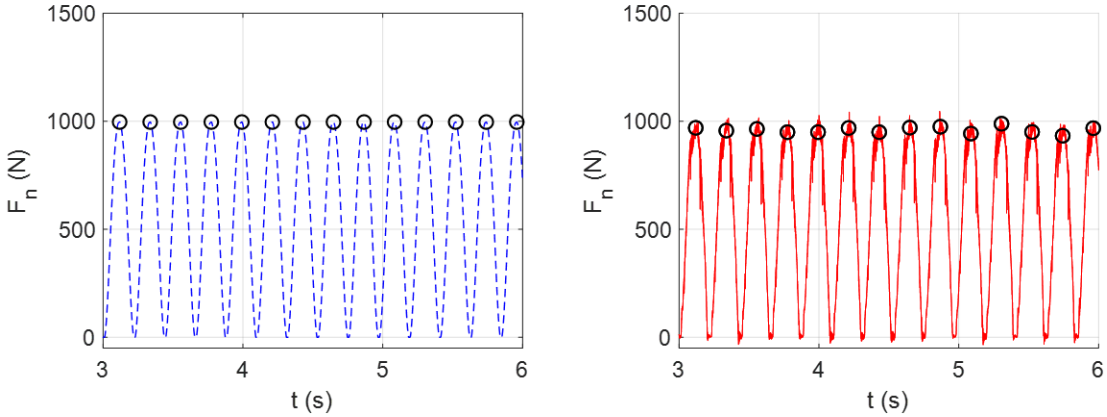

Figure 6.6: Predicted and measured cutting force in the normal direction for test point A. (Left) predicted, (right) measured.

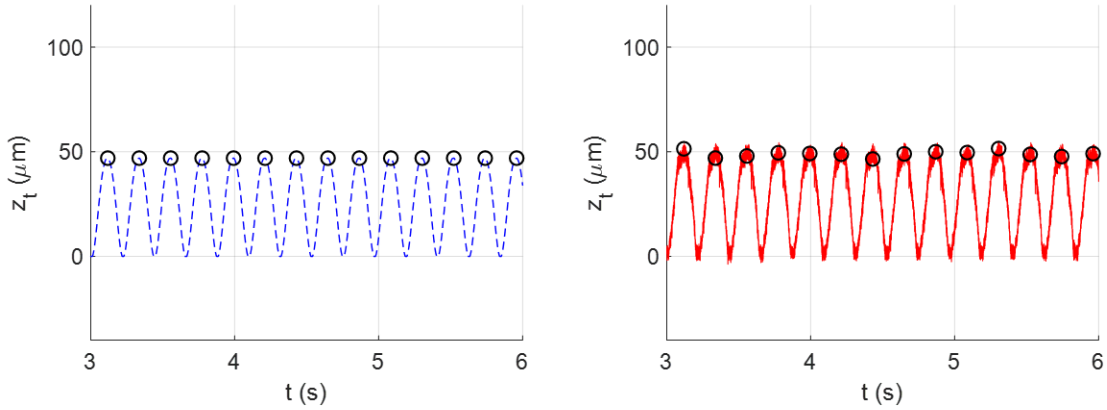

Figure 6.7: Predicted and measured tool displacement for test point A. (Left) predicted, (right) measured.

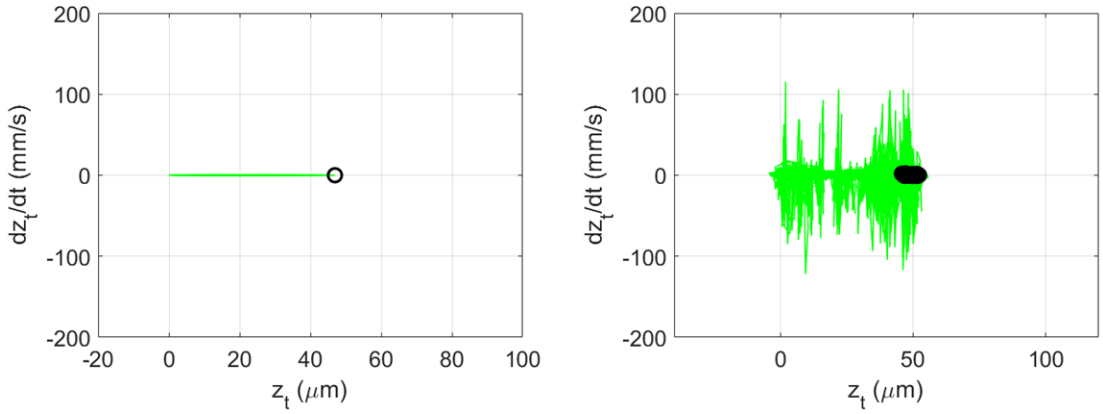

Figure 6.8: Predicted and measured Poincaré map for test point A. (Left) predicted, (right) measured.

Due to the sampled points remaining a constant value point to point for the time domain signals and the sampled points of the Poincaré map repeating around a single point, the cut was determined to be a stable (non-chatter) cut. The determination of cutting stability matches the result of the global stability map shown in Figure 6.1. Matching the predicted and measured signals of a stable cutting case indicates that the cutting force model is an appropriate model, thus allowing the tool displacement to be used to determine the cutting stability of future tests.

In Figure 6.9 and Figure 6.10, the tool displacement and Poincaré map is provided for test point B (b = 4.5 mm, RAF = 0.64, OPR = 1.54). Figure 6.9 shows the predicted and measured tool displacement. The periodic sampling points are shown by the dark circles and are overlaid on the tool displacement signals. Figure 6.10 shows the predicted and measured Poincaré map. Both the predicted and measured Poincaré maps match each other closely with minimal differences between the sampled points. Due to the sampled points changing value point to point in the time domain signals and the sampled points of the Poincaré map creating an ellipse (indicative of a secondary Hopf bifurcation), the cut was determined to be unstable (chatter). The determination of unstable cutting matches the result of the global stability map shown in Figure 21. By matching the predicted and measured signals transition from stable to unstable cutting, this indicates that the tool dynamics were appropriately fitted.

In Figure 6.11 and Figure 6.12, the tool displacement and Poincaré map is provided for test point C (b = 4.5 mm, RAF = 0.875, OPR = 0.492). The predicted

and measured tool displacement is shown in Figure 6.11, whereas the predicted and measured Poincaré maps are shown in Figure 6.12. Agreement between predicted and measured signals are present. Test point D was determined to be an unstable cut due to the formation of an ellipse with the sampled points of the Poincaré map. This agrees with the global stability map shown in Figure 6.2.

In Figure 6.13 and Figure 6.14, the tool displacement and Poincaré map is provided for test point D (b = 4.5 mm, RAF = 0.89, OPR = 0.593). Figure 6.13 and Figure 6.14 both show agreement between predicted and measured tool displacement and Poincaré maps, respectively. Test point D was determined to be an unstable cut due to the variation of sampled points in the time domain signal. This agrees with the global stability map shown in Figure 6.2.

In Figure 6.15 and Figure 6.16, the tool displacement and Poincaré map is provided for test point E (b = 4.5 mm, RAF = 1.6, OPR = 0.496). Figure 6.15 shows the predicted and measured tool displacement. Both the predicted and measured signals have the show good agreement in both amplitude and in tool displacement profile shape. Figure 6.16. shows the similarity between the predicted and measured Poincaré maps. The cutting test was determined to be a stable cut, which agrees with the global stability map (Figure 6.2).

Summary: Tube turning (orthogonal turning) tests were conducted to compare the simulation results to the measured signals to ensure model accuracy. Good agreement was observed between predicted and measured signals for several cutting test with varying cutting and MTP parameters ({RAF,OPR} combinations). Cutting stability was controlled by only changing MTP paraments.

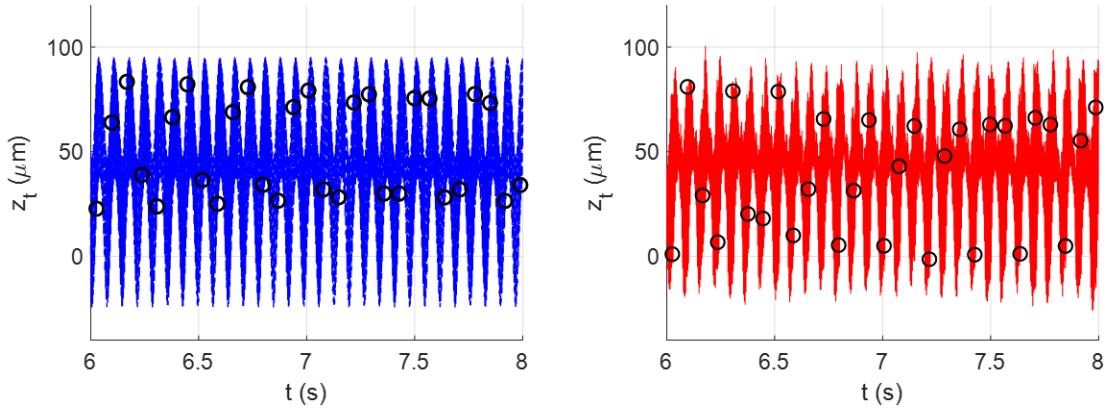

Figure 6.9: Predicted and measured tool displacement for test point B. (Left) predicted, (right) measured.

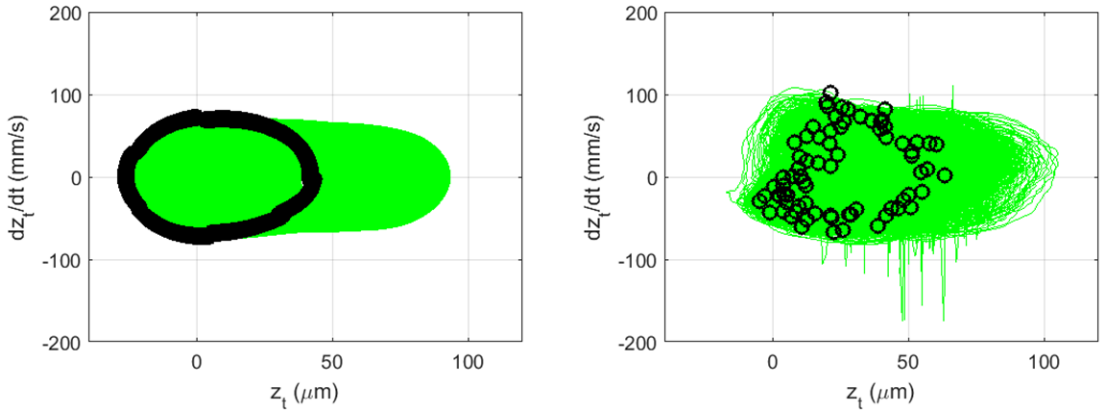

Figure 6.10: Predicted and measured Poincaré map for test point B. (Left) predicted, (right) measured.

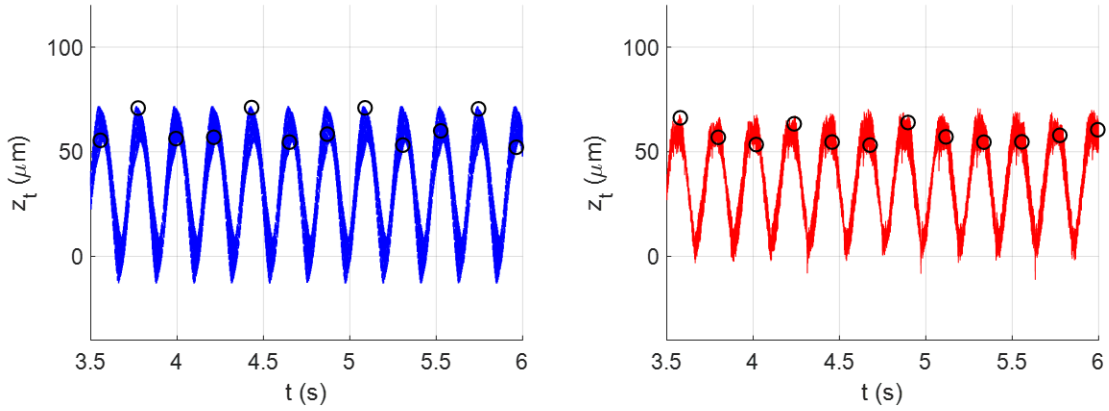

Figure 6.11: Predicted and measured tool displacement for test point C. (Left) predicted, (right) measured.

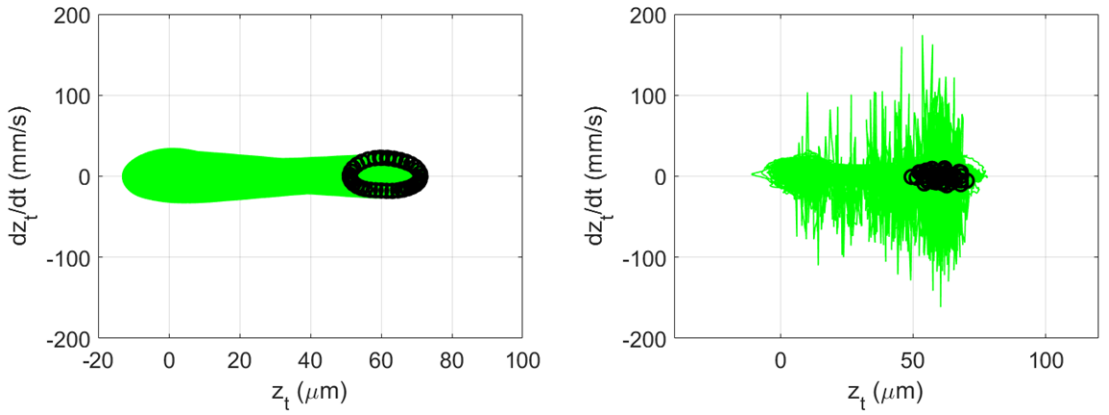

Figure 6.12: Predicted and measured Poincaré map for test point C. (Left) predicted, (right) measured.

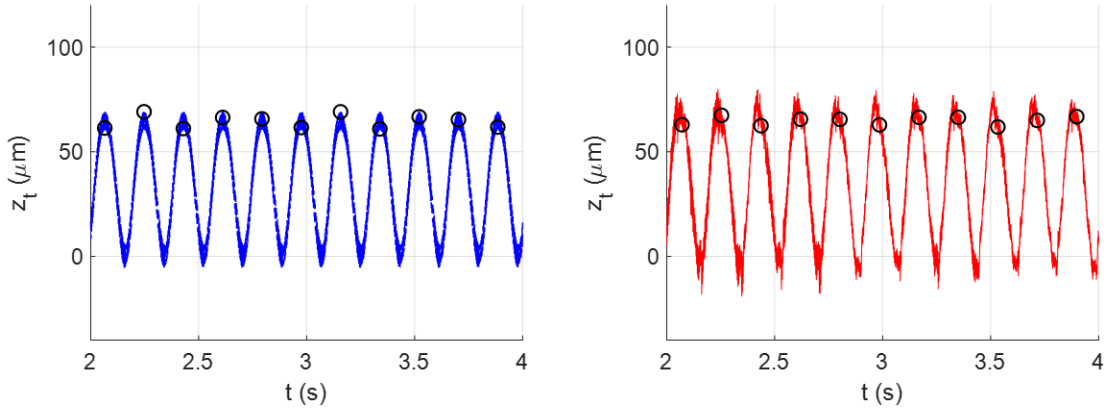

Figure 6.13: Predicted and measured tool displacement for test point D. (Left) predicted, (right) measured.

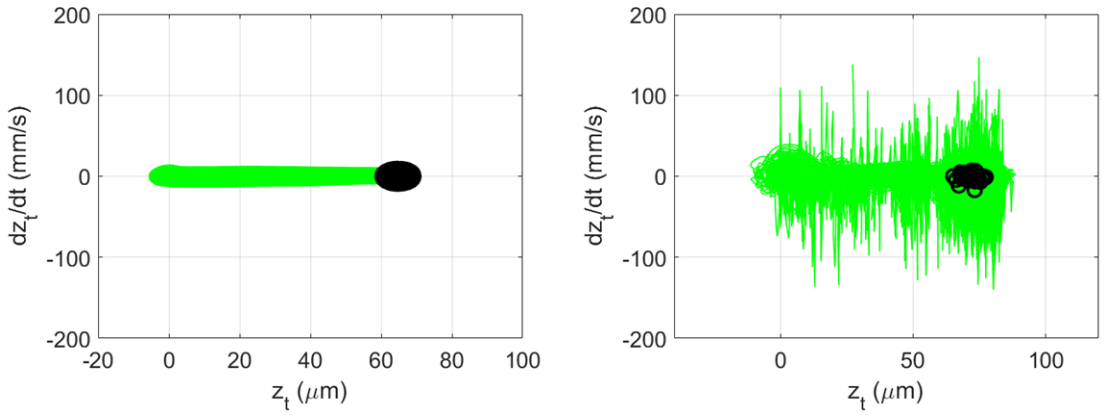

Figure 6.14: Predicted and measured Poincaré map for test point D. (Left) predicted, (right) measured.

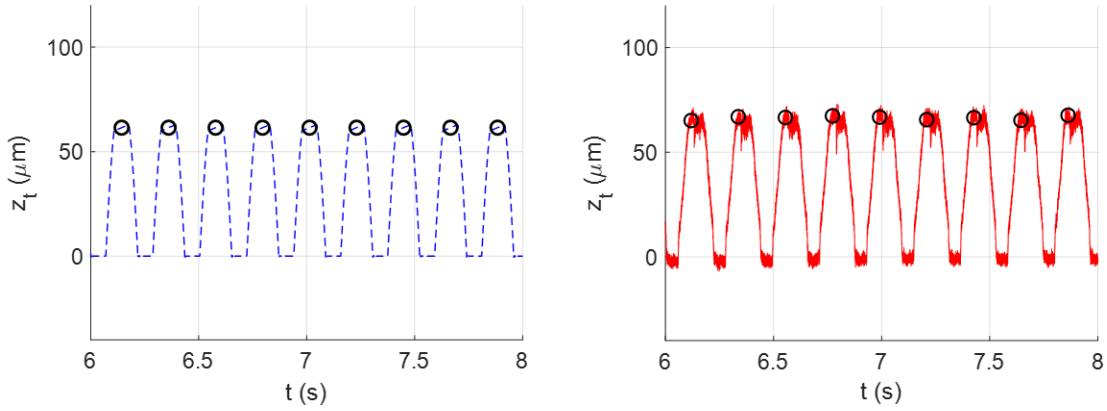

Figure 6.15: Predicted and measured tool displacement for test point E. (Left) predicted, (right) measured.

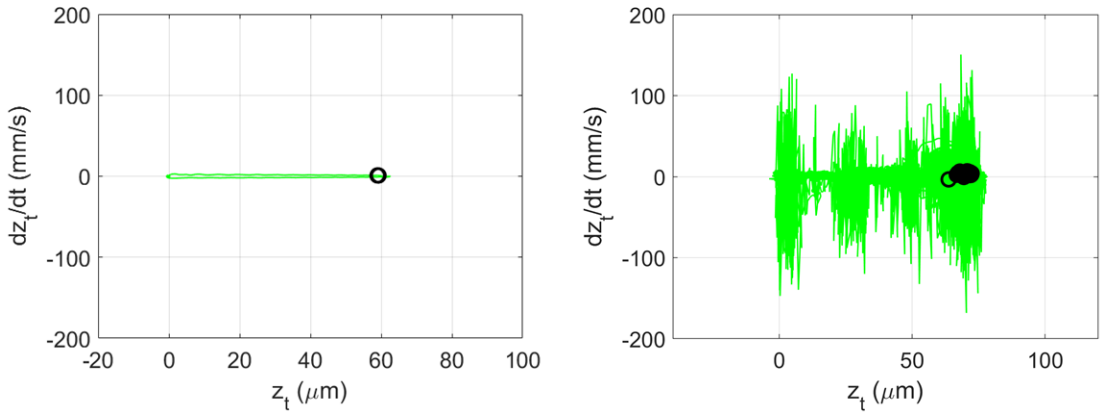

Figure 6.16: Predicted and measured Poincaré map for test point E. (Left) predicted, (right) measured.

MTP surface finish

Using the setups described in Figure 5.5 and Figure 5.6, a study to investigate the effect of MTP parameters {RAF and OPR } pairs on the generated surface profile was completed. Using the time domain simulation described in Chapter Four, simulations were carried out to compare the predicted and measured surface profiles. The effects of plastic side flow on the effective tool nose radius [39] was observed early in testing. An effective nose radius model was therefore implemented to predict surface profiles more accurately.

Effective nose radius and modeling

To find the effective nose radius, a constant feed (traditional turning) cutting test was conducted at a commanded chip thickness of 0.051 mm/rev (the same global feed rate as the planned MTP cutting tests). Once the cutting test was completed, the test workpiece's surface profile was measured using the Mitutoyo Contracer shown in Figure 5.9. Then using the time domain simulation, the nose radius was varied until the predicted surface profile matched the measured cusp pattern left behind by the tool during the cutting process. This enabled the effective nose radius of 0.234 mm to be identified and used when solving Eq. 23 in the time domain simulation. Figure 6.17 shows the plot resulting from the effective nose radius fitting exercising. Good agreement is observed between the predicted and measured surface profiles using the effective nose radius instead of the physical insert nose radius (0.397 mm).

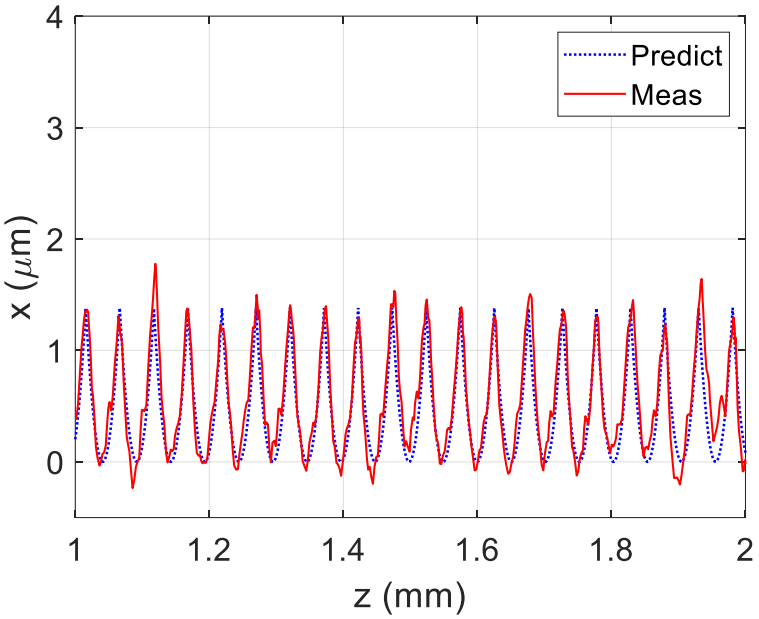

Figure 6.17: Effective nose radius extraction by fitting the predicted profile to the measured profile for a constant feed cutting test with a commanded chip thickness of 0.051 mm.

Cutting test parameter locations

Figure 6.18 displays the surface finish cutting test locations plotted on the chip breaking map. A central test case of RAF = 0.8 and OPR = 0.5 is denoted by a black dot. The horizontal blue line represents the varied RAF values, while the vertical red line represents the varied OPR values. For both the blue and red lines, the circles indicate the location of a cutting test RAF and OPR pair. The desired, commanded, and measured MTP parameters are listed in Table 6.2. The measured MTP parameters were verified using the laser triangulation sensor shown in Figure 5.5. The analysis procedure used to verify the commanded tool paths was similar to the process used to verify the cutting stability tool paths. At each test location, a time domain simulation generated surface profile was overlayed and compared to the measured cutting test surface profile. The R_a (see Eq. 24) was also calculated for both predicted and measured surface profiles.

Issues while conducting experiments

When verifying the commanded tool paths by cutting air and measuring the resulting tool displacement, an excessive vibration was observed. The amplitude of vibration was so large that the mounting bracket that connects the machine tool's controller to the machine base was excited and began to vibrate with a large amplitude. Several possible root causes were investigated, such as one of the machine tool's feet not being firmly mounted to the concrete floor or instability in the machine's controller response while trying to perform the commanded MTP moves.

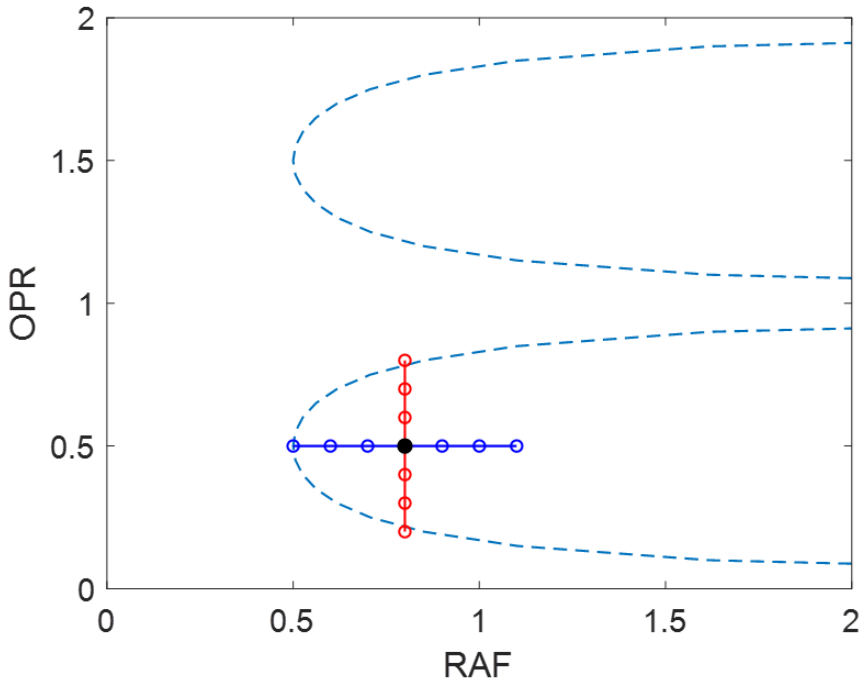

Figure 6.18: The central point with an RAF and OPR of 0.8 and 0.5, respectively, is shown by the solid black dot (superimposed on Figure 1.5). The red line denotes OPR varied points, the blue line denotes RAF varied points, and open circles represent individual test locations.

Table 6.2. Surface finish cutting tests naming convention. The desired, commanded, and measured $\{RAF, OPR\}$ pairs are provided for each cutting test.

Test name	Desired		Comanded		Measured	
	RAF	OPR	RAF	OPR	RAF	OPR
O31142	0.8	0.2	1.9	0.2	0.8	0.205
O31242	0.8	0.3	1.9	0.3	0.8	0.305
O31342	0.8	0.4	1.85	0.39	0.8	0.394
O31442	0.8	0.5	1.9	0.502	0.8	0.509
O31542	0.8	0.6	2.5	0.59	0.8	0.602
O31642	0.8	0.7	3	0.69	0.8	0.71
O31742	0.8	0.8	3.5	0.79	0.8	0.81
O31412	0.5	0.5	1.5	0.502	0.5	0.511
O31422	0.6	0.5	1.65	0.502	0.6	0.509
O31432	0.7	0.5	1.85	0.502	0.7	0.505
O31452	0.9	0.5	1.97	0.502	0.9	0.509
O31462	1.0	0.5	2.25	0.502	1.0	0.509
O31472	1.1	0.5	2.5	0.502	1.1	0.512

However, these possible issues were eliminated. Eventually, the root cause of the vibration issue presented itself as the machine tool's carriage (z-axis) handle. The handle had enough play in the gearing system that the outer diameter MTP turning oscillations caused the carriage's handle wheel to violently vibrate back and forth due to the offset mass (handle) from the center of the wheel. Figure 6.19 displays a photograph of the machine tool carriage wheel's offset handle. To resolve the excessive vibrations issue, the carriage handle wheel was removed while cutting tests were being conducted; this eliminated the violent vibrations observed in early testing.

The second issue that arose when matching the measured and predicted surface profiles was the issue of automatic filtering, form removal, averaging, and other data analysis techniques settings that were unable to be turned off when using a handheld profilometer to collect the surface profile data. This posed an issue because the surface profile that was exported from the handheld profilometer was not representative of the actual surface profile trace. Ideally, the raw data should be collected and filtering, form removal, averaging, and other surface data processing techniques should only be applied when needed to remove unwanted surface content. Otherwise, inappropriate changes are made to the surface profiles. This issue was the motivation behind using the Mitutoyo Contracer detailed in Figure 5.9. Using the Contracer, raw surface profiles were collected, and post analysis was completed in Matlab. The only surface data processing techniques employed for this study was a linear fit (slope) removal and a high pass spatial frequency filter.

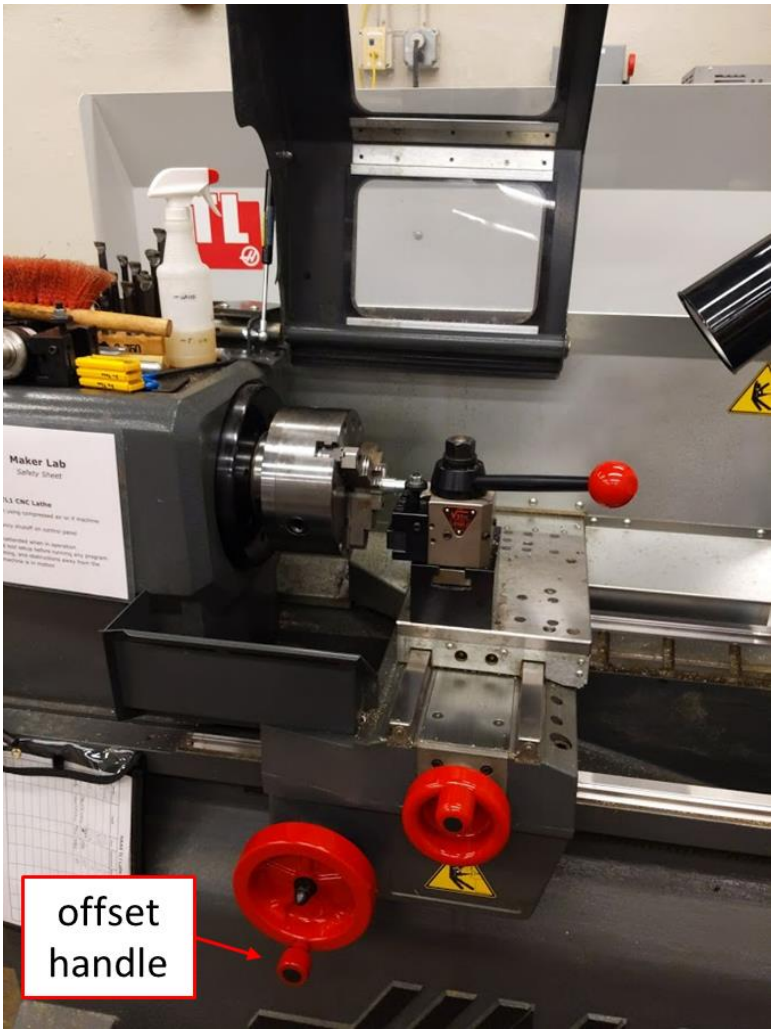

Figure 6.19: Photograph of the Haas TL-1's carriage offset handle.

The linear fit was applied to remove any tilt in the surface profile data resulting from small misalignments between the profilometer's reference surface and the surface of the workpieces. The high pass spatial frequency filter was applied to remove any large wavelength surface content (such as waviness) on the test sample surface profiles. A high pass fourth order Butterworth filter with a spatial frequency cutoff of 0.8 mm^{-1} was used to facilitate the high pass spatial frequency filter. Applying this filter to the surface profile removes any surface content with a wavelength larger than 1.25 mm. Figure 6.20 and Figure 6.21 show the surface data processing technique detailed above that was applied to each surface profile collected in this study.

Cutting test results

The results section is separated into two subsections that each review surface profile shape and R_a value changes with MTP operating parameters. The first subsection reviews the effects of varying the OPR value. The second subsection reviews the effects of varying the RAF value.

– *OPR variation*

To investigate the effect of the OPR value on the surface profiles, the following cutting test were employed: O31142, O31242, O31342, O31442, O31542, O31642, and O31742. The cutting test locations are shown visually in Figure 6.18 by the red line with circles that denote the individual cutting test locations. Figure 6.22 shows a predicted and measured surface profile overlay for the test location O31242 where RAF = 0.8 and OPR = 0.3.

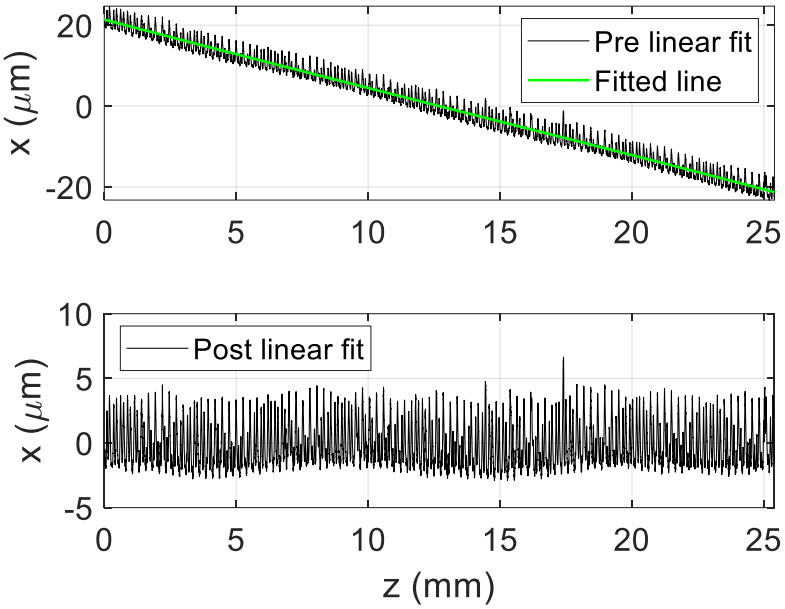

Figure 6.20: Example linear fit slope removal of a surface profile. The black line represents the surface profile data while the green line represents the linear fitted line. (Top) pre slope removal, (bottom) post slope removal.

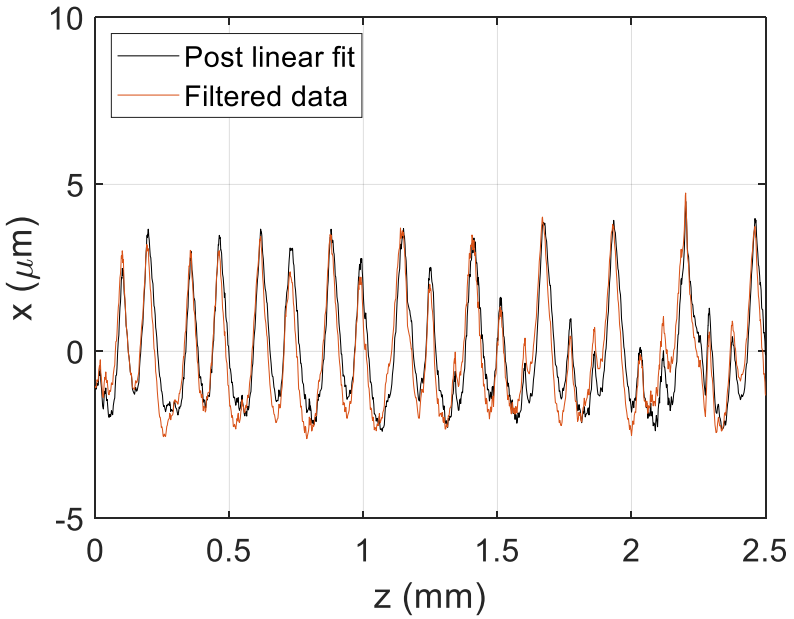

Figure 6.21: Example high pass spatial frequency filter applied to a measured surface profile. The black line represents the pre filtered data while the orange line represents the post filtered data.

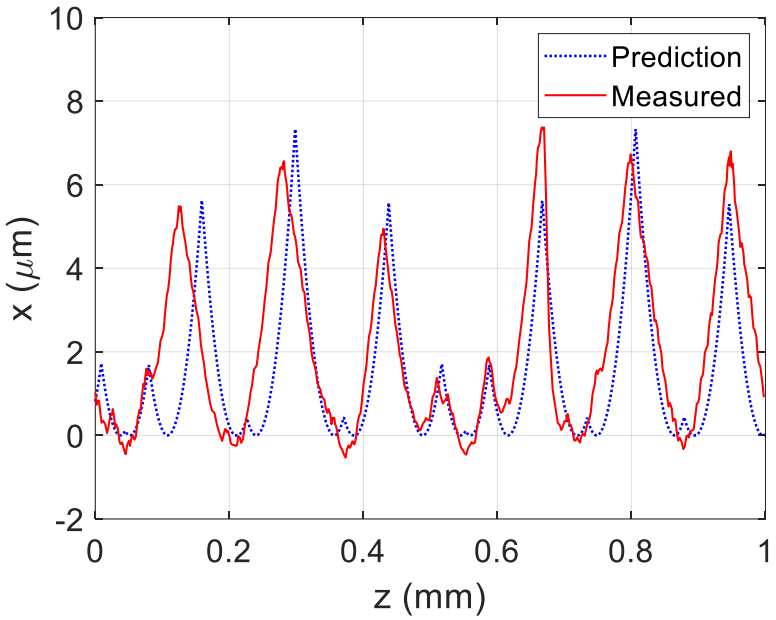

Figure 6.22: Surface profile comparison for test point O31242, repeat testing location at jaw 1. The measured MTP parameters were RAF = 0.8, OPR = 0.3.

Good agreement between predicted and measured surface profiles is observed for both the general profile shape and profile amplitude.

The superposition of predicted and measured surface profiles exercise is repeated for two other locations (approximate equal radial spacing of 120 degrees) on the same test workpiece cut surface. This was done to demonstrate the surface profile behavior is the same at different circumferential locations on the test workpiece. The approximate 120 degree spacing was achieved by marking each location of the lathe's three-jaw-chuck jaw locations on the test workpiece and measurements were carried out at each location. Figure 6.22 - Figure 6.24 are provided to show the comparison between predicted and measured surface profiles at the different circumferential (three-jaw-chuck jaw) locations on the test workpieces cut surfaces. Good agreement between the predicted and measured surface profiles is observed for Figure 6.22 - Figure 6.24 indicating that measurements made at one location along the length of the workpiece is representative of the entire test surface.

Figure 6.25 shows the calculated R_a values for both the measured and predicted surface profile as the OPR value is varied. An interesting relationship between the R_a and OPR values are observed in the figure. The test points with lower OPR values are mirrored about the central test case (O31442, RAF = 0.8, and OPR = 0.5) as seen by similar values for the larger OPR value test cases. This relationship is due to the geometry of the commanded tool path being mirrored about the central test case.

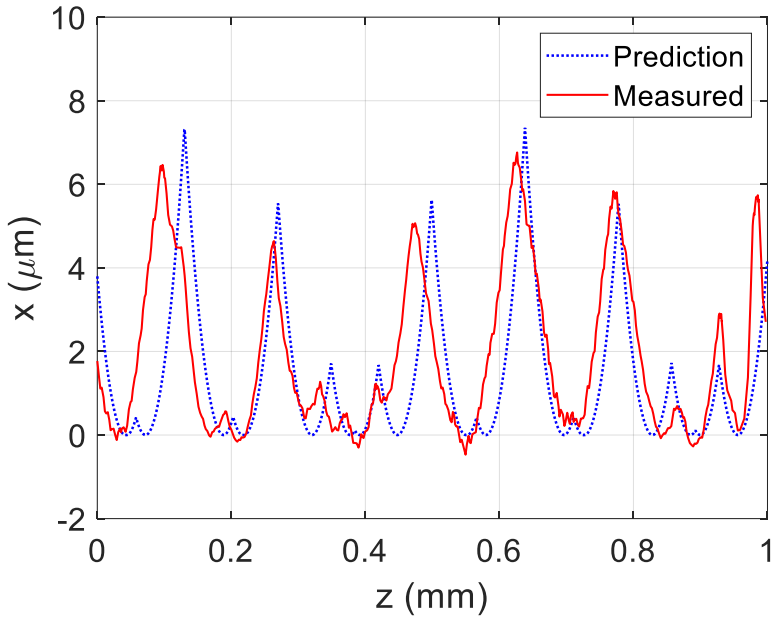

Figure 6.23: Surface profile comparison for test point O31242, repeat testing location at jaw 2. The measured MTP parameters were RAF = 0.8, OPR = 0.3.

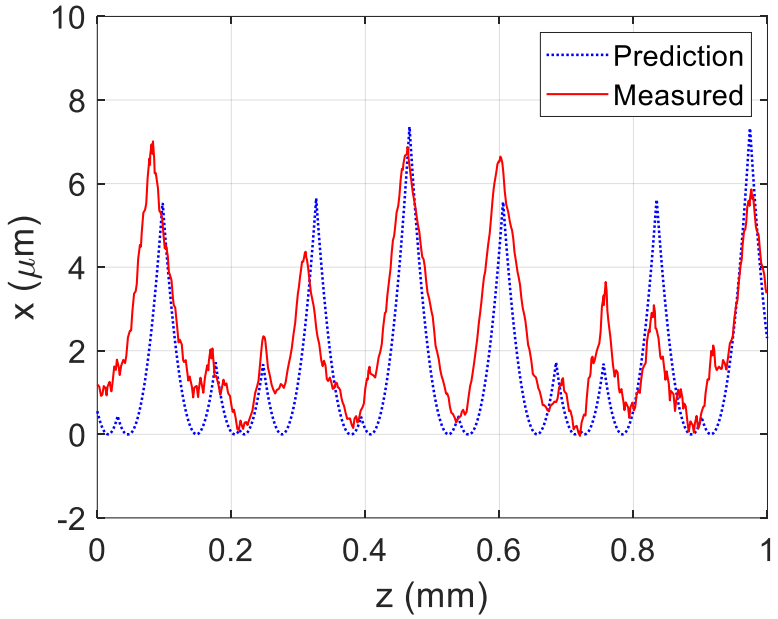

Figure 6.24: Surface profile comparison for test point O31242, repeat testing location at jaw 3. The measured MTP parameters were RAF = 0.8, OPR = 0.3.

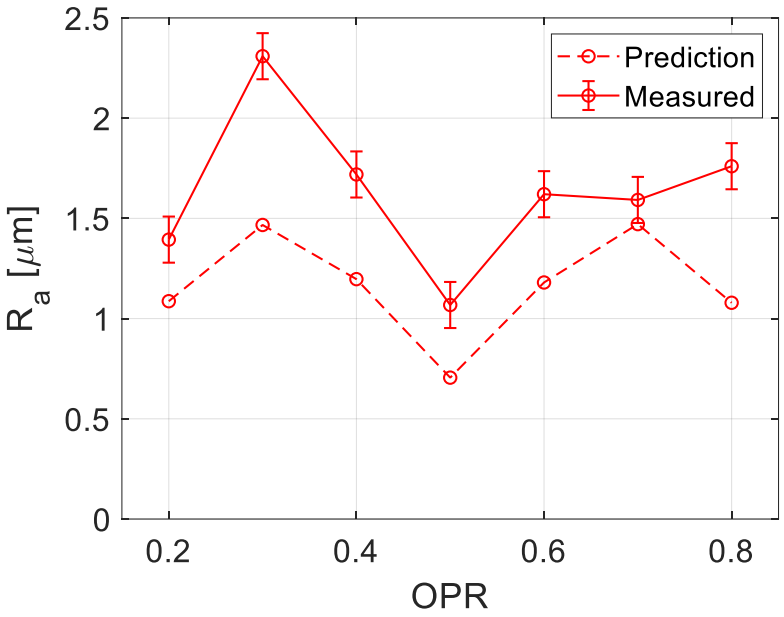

Figure 6.25: Arithmetic mean surface roughness (R_a) value as OPR is varied. Plotted error bars represent one standard deviation.

There is an offset difference between the predicted and measured R_a values which is due to the higher spatial frequency surface content that was not modeled in the time domain simulation. Otherwise the general shape is similar between the predicted and measured R_a values. The good agreement between prediction and measurement indicates that the trend of R_a values was captured by the time domain simulation.

The reader is encouraged to view the additional overlaid surface profile comparison plots located in Additional predicted and measured surface profiles subsection OPR variation. Figure A.1 - Figure A.7 further validate that accurate predictions of surface profiles was achieved by showing the fitted surface profile for the individual tests that make up Figure 6.25.

– *RAF variation*

To investigate the effect of the RAF value on the surface profiles, the following cutting test were employed: O31412, O31422, O31432, O31442, O31452, O31462, and O31472. The cutting test locations are shown visually in Figure 6.18 by the blue line with circles that denote the individual cutting test locations. Figure 6.26 shows a predicted and measured surface profile overlay for the test location O31422 where the MTP parameters were RAF = 0.6 and OPR = 0.5. Good agreement between predicted and measured surface profiles is observed for both the general profile shape. The surface profile amplitude differs only slightly.

To ensure the surface profile was consistent at different circumferential locations for the same workpiece, the same procedure was applied as in the OPR varied test points analysis. Figure 6.26 - Figure 6.28 are provided to show the comparison between prediction and measured surface profiles at the different circumferential (three-jaw-chuck jaw) locations. Good agreement between the predicted and measured surface profiles is observed for Figure 6.26 - Figure 6.28 indicating that a single measurement is representative of the whole test surface.

Figure 6.29 shows the calculated R_a values for both measured and predicted surface profiles as the RAF value is varied. A power law relationship is observed that shows that as the RAF value is increased, the calculated R_a values decreases. As observed in the OPR varied points analysis, an offset is present where the predicted R_a values are lower than the measured R_a values. As in the previous analysis, this is due to the higher spatial frequency surface content that was not modeled in the time domain simulation. Otherwise the general shape is similar between the predicted and measured R_a values. The good agreement between prediction and measurement indicates that the trend of R_a values was captured by the time domain simulation.

The reader is encouraged to view the additional overlaid surface profile comparison plots located in Additional predicted and measured surface profiles subsection RAF variation. Figure A.8 - Figure A.14 further validate that accurate predictions of surface profiles was achieved by showing the fitted surface profile for the individual tests that make up Figure 6.29.

Summary: Outer diameter finish turning tests were conducted to compare the time domain simulation results to the measured surface profiles. Several issues that arose during early testing were presented and methods to resolve the issues were discussed. Good agreement when matching surface profiles and $R_a$values for predicted and measured profiles was observed. The MTP parameters effect on the surface profile shape was captured for multiple {RAF, OPR} pairs.

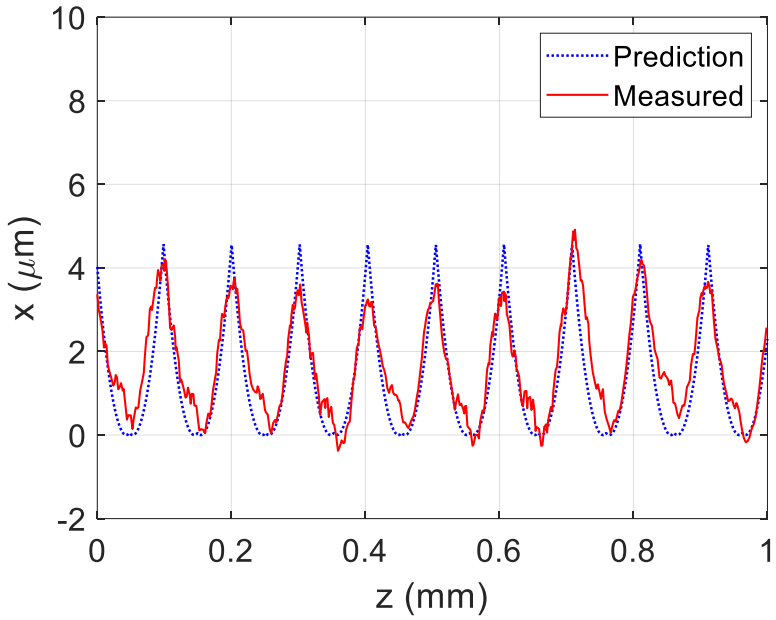

Figure 6.26: Surface profile comparison for test point O31422, repeat testing location at jaw 3. The measured MTP parameters were RAF = 0.6, OPR = 0.5.

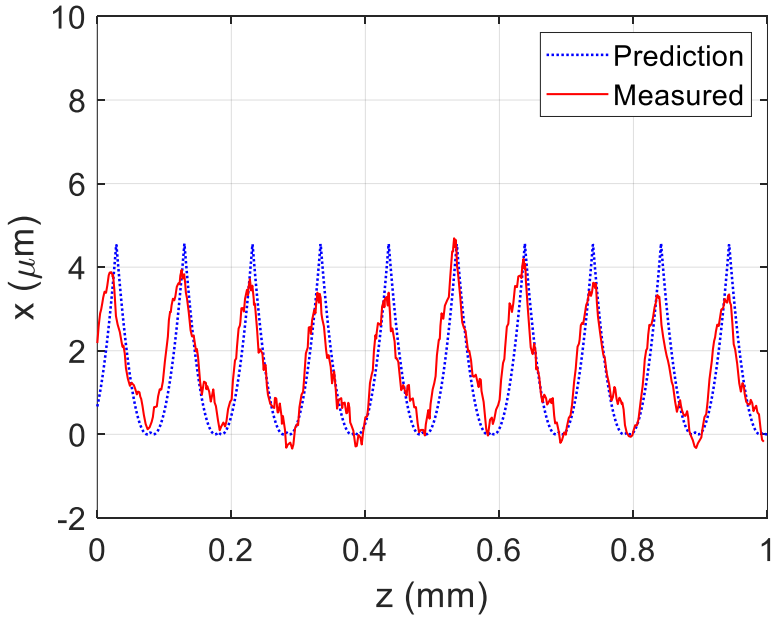

Figure 6.27: Surface profile comparison for test point O31422, repeat testing location at jaw 3. The measured MTP parameters were RAF = 0.6, OPR = 0.5.

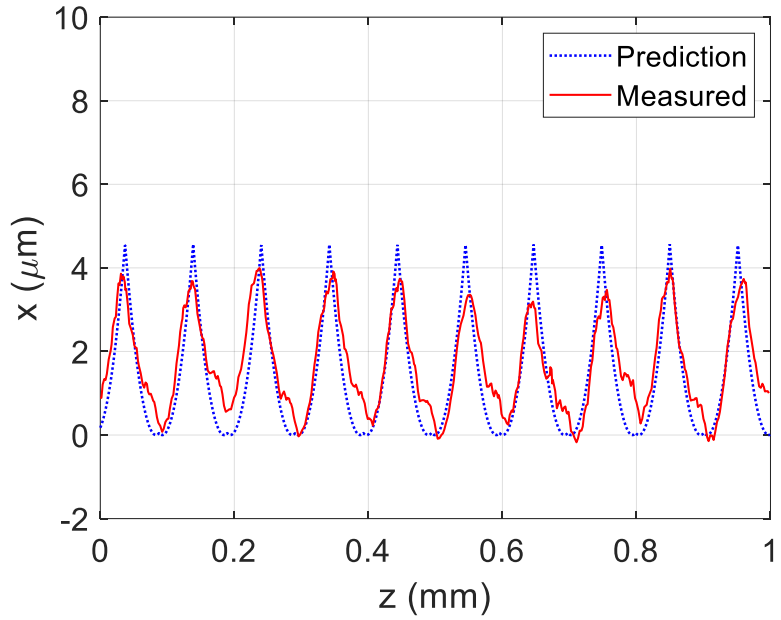

Figure 6.28: Surface profile comparison for test point O31422, repeat testing location at jaw 3. The measured MTP parameters were RAF = 0.6, OPR = 0.5.

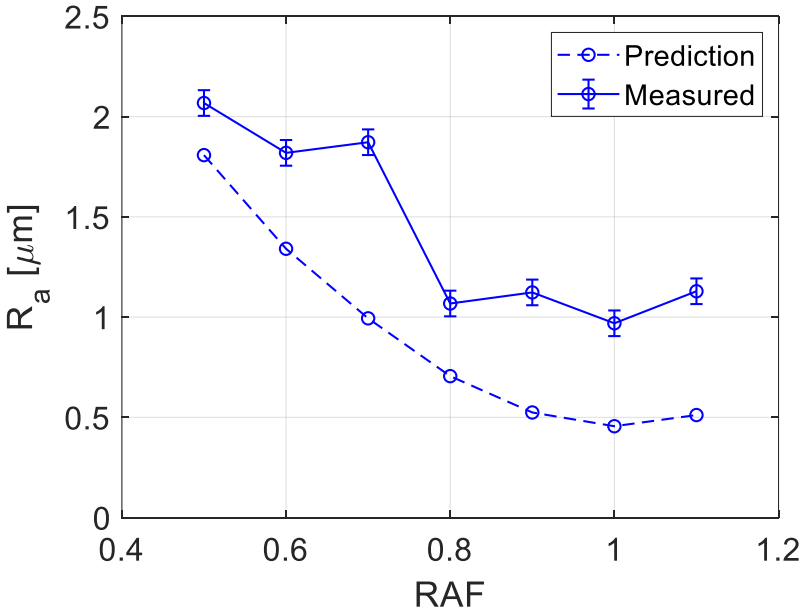

Figure 6.29: Arithmetic surface roughness (R_a) value as RAF is varied. Plotted error bars represent one standard deviation.

CHAPTER SEVEN
CONCLUSIONS AND FUTURE WORK

The research presented in this dissertation addressed several issues that have not been resolved by previous work in the area of MTP and MAM machining. A primary contribution to the state of the art is the derivation, programming, and validation of a numerical simulation that is able to predict cutting stability for MTP turning. This provides manufacturing engineers and machine tool programmers the ability to predict the optimal machining parameters to maximize the material removal rate. Thus, the amount of scrapped parts due to unacceptable surface finish resulting from unstable cutting is minimized, tool damage due to excessive forces is avoided, and the ‘spindle on time’ of the machining process is reduced.

Another contribution is the derivation, programming, and validation of a comprehensive surface finish numerical simulation that includes the tool and workpiece structural dynamics and plastic side flow effects. This provides researchers, manufacturing engineers, and machine tool programmers the ability to predict the appropriate MTP parameters to meet surface finish requirements. With the ability to predict the surface finish ahead of time and off site of the machine tool, the number of test parts required to approve a product manufacturing process is reduced. This also reduces the amount of material and tooling dedicated only to testing, mitigates increased machine down time, and decreases machine setup time.

Conclusions

During this study, a numerical simulation to solve the second-order, time-delay differential equations of motion that describe the MTP process dynamics was developed. The simulation output signals (force, displacement, and velocity) were analyzed using a periodic sampling method to identify stable and unstable behavior (i.e., forced vibration or secondary Hopf bifurcation). A metric that analyzes the periodic sampled points was employed to enable automatic stability determination. The metric enabled stability maps to be generated without need of human determination of stability which is very time consuming.

An experimental in-process metrology setup that approximates orthogonal (tube) turning was assembled to validate the time domain generated stability maps. In-situ measurements of the normal direction cutting force, displacement, velocity, and spindle speed signals were completed and analyzed post-process to determine cutting stability. Using the numerical simulation along with the measured frequency response functions and cutting force coefficient model, stability maps for the experimental setup were generated.

Tube turning tests were conducted to compare the simulation results to the measured signals to ensure model accuracy. Good agreement was observed between predicted and measured signals for multiple cutting tests while varying cutting and MTP parameters ({RAF,OPR} combinations). It was observed that the cutting stability was controlled by only changing MTP paraments. It was also

observed that typically as the RAF value is increased, the cutting behavior would transition from unstable (chatter) to stable cutting conditions.

Future work

From this study, the MTP stability map (see Figure 6.2) was investigated mainly in the stable lower lobe. Next steps would be to conduct a further investigation into the upper two lobes of the MTP stability plots to see verify that stable cutting zone exist. This would require a more responsive machine tool than the Haas TL-1 lathe shown in Figure 5.1 to facilitate the higher OPR values which result in higher oscillation frequencies.

One of the complications during this study was the high computational cost of generating the MTP stability map. An area for future work is to apply knowledge learned from this study to reduce the time required to generate the maps. From the MTP stability map, it was shown that typically if you had a cutting test that exhibited unstable cutting behavior, increasing the RAF value would result in the transition to stable cutting. This presents the opportunity to add logic to the generation of the maps to cut down on the require simulation time. When the numerical simulation is rastering across the grid of points (see Figure 3.7) and a previously unstable simulated cut transitions to a stable cut due to increasing the RAF value it can be inferred that the rest of the row of increasing RAF values will also be stable. This trick could be implemented after n number of grid points to ensure that a false positive determination of stable cutting would occur. This would

dramatically reduce the required simulation time. In the case of Figure 6.2, it would remove almost a third of the required simulation grid points.

The G-code that is generated to command the machine tool to execute the desired MTP motion is made up of a six-point approximation of the sinusoidal oscillations imposed on the global feed rate [5-6]. This size of the code can become increasingly large for larger commanded lengths of cut and the code can therefore become cumbersome to edit. A possible improvement would be to implement a macro program where the machine operator calls a custom G-code that performs the tool path calculation on the controller. The only inputs would be the commanded RAF and OPR pair, global feed rate, and the commanded position to for the axes. The macro would then behave in a similar manner to a linear interpolation positioning command (G01) with the addition of automatic chip breaking. This has several advantages. First, when troubleshooting an MTP program, if the measured parameters are not acceptable, the machine operator can simply change the macro input parameters without having to re-post the MTP code and upload it to the controller. Second, the macro program could be used as part of a chatter suppression tool. Using an in-situ cutting stability detection system (such as a microphone), when chatter is detected, the macro variables could be updated to increase the commanded RAF value until the cutting test transitions to stable cutting behavior.

Conclusions

Starting with the stability prediction simulation, the following additions were included in the simulation: the radial tool and workpiece dynamics were added along with the process to predict the surface finish and calculate the arithmetic mean surface roughness, R_a. The revised numerical simulation output signals included the profile height and trace length that make up a predicted surface profile trace. The ability to account for an effective nose radius resulting from the effects of plastic side flow due to small instantaneous chip thicknesses was modeled.

An outer diameter finish turning testing setup that included in-process metrology to capture the actual tool motion was developed. Post-process profilometry was provided using a Mitutoyo Contracer. Using the numerical simulation along with the measured tool and workpiece frequency response function and cutting force model, surface profiles were predicted at various cutting conditions for comparison to measured surface profiles.

Outer diameter finish turning cutting tests were conducted to compare the simulation results to the measured signals to confirm model accuracy. Good agreement was observed for multiple tests at different locations around the circumference of the same test workpieces. Comparisons between the predicted and measured R_a values for various MTP parameters was made and the general trends was captured by the numerical simulation. Additional surface profile comparisons were made that demonstrated the numerical simulation's ability to

predict the measured surface profile shape and amplitude for multiple RAF and OPR pairs.

Future work

In this study the ability to predict the surface profile for an outer diameter finish turning cutting test was demonstrated. The cutting conditions are similar to inner diameter (boring) finish turning conditions thus the existing numerical simulation should remain unchanged to model internal surface profiles. The next step would be to conduct inner diameter finish turning cutting tests to compare to the numerical simulation to validate model accuracy and make any additional changes to the numerical simulation if necessary. To further enable additional modeling callabilities, the numerical simulation could be modified to account for combined axis turned parts.

When identifying the plastic side flow model, physical constant feed (traditional) turning tests at various commanded chip thickness was required. Sometimes conducting cutting tests is not possible due to limited availability of the workpiece material or cutting inserts. Another issue is that with global markets sometimes the researcher is not located where the machine tool is and does not have access a machine tool to perform the cutting tests. A possible solution is to investigate methods to estimate the plastic side flow model using material dependent properties such as the Johnson-Cook flow stress parameters.

When comparing the predicted and measured surface profiles (see Figure 6.25 and Figure 6.29) the predicted R_a values were lower than the measured R_a

values. This was due to the higher spatial frequency surface content that was not modeled in the numerical simulation. Future work could include investing the physics that cause the higher spatial frequency surface content shown in Figure 6.22 - Figure 6.24 and Figure 6.26 - Figure 6.28. The addition of the higher spatial frequency surface content physics to the numerical simulation is postulated to reduce the discrepancy between the predicted and measured R_a values.

LIST OF REFERENCES

[1] Zhang, Y.Z, 1980, Chip Curl, Chip Breaking, and Chip Control of Difficult-to-cut Materials, Annals of the CIRP, 29/1: 79-83.
[2] Rasch, F.O., and Vigeland, T., (1981). “Hydraulic Chipbreaking”, Annals of the CIRP, Vol. 30, pp. 333-335
[3] Woodbury, Robert S. "The Origins of the Lathe." Scientific American 208, no. 4 (1963): 132-43. Accessed July 19, 2020. www.jstor.org/stable/24936538.
[4] Adams, D., 2008, Chip Breaking in Turning Operations Using CNC Toolpaths, MS Thesis, UNC Charlotte, Charlotte, NC.
[5] Assaid, T.S., 2010, Generation, Measurement, and Assessment of Modulated Tool-Path Chip Breaking in CNC Turning Processes, MS Thesis, UNC Charlotte, Charlotte, NC.
[6] Tursky, D., 2010, Chip Breaking Through Oscillating CNC Toolpaths and its Effect on Chip Length, Tool Wear and Machine Dynamics, MS Thesis, UNC Charlotte, Charlotte, NC.
[7] Woody, B.A., Smith, K.S., Adams, D.J. Barkman, W.E., 2008, Assessment of the Process Parameters and their Effect on the Chip Length When Using CNC Toolpaths to Provide Chip Breaking in Turning Operations, Proceedings of the 3rd ASME MSEC Conference, USA.
[8] Woody, B.A., Smith, K.S., Adams, D.J., Barkman, W.E., 2008, “Chip Breaking in Turning operations in CNC Toolpaths” Transactions of NAMRI, 36:1-8.
[9] Woody, Bethany A., Smith, Kevin Scott, Adams, David J., Barkman, William E., Babelay, Jr., Edwin F., Methods and systems for chip breaking in turning applications using CNC toolpaths, U.S. Patent 8 240 234, issued August 14, 2012.
[10] Smith, S., Barkman, W., Woody, B., and Adams, D., 2008, “Assessment of the Process Parameters and Their Effect on Chip Length When Using CNC Toolpaths to Provide Chip Breaking in Turning Operations”, Proceedings of the 3rd International Conference on Manufacturing Science and Engineering (MSEC)
[11] Smith, S., Barkman, W., Babelay, E., Tursky, D., and Assaid, T., 2009, “Machine Tool Performance Requirements with Modulated Tool-Path Chip Breaking”, Proceedings of the ASPE 2009 Annual Conference
[12] Smith, S., Woody, B., Barkman, W., and Tursky, D., 2009, “Measurement of Tool Temperature Using Thermal Imaging in Interrupted Cut Turning”, ASME International Manufacturing Science and Engineering Conference (MSEC)
[13] Smith, S., Woody, B., Barkman, W., and Tursky, D., 2009, “Temperature Control and Machine Dynamics in Chip Breaking using CNC Toolpaths”, accepted for the Annals of the CIRP, Vol. 58.

[14] Smith, S., Barkman, W., Babelay, E., Tursky, D., McFarland, J., and Assaid, T., 2010, “Surface Characteristics Generated in CNC Chip Breaking Tool Paths”, accepted for the Annals of the CIRP

[15] Smith K.S., McFarland J.T., Tursky D.A., Assaid T.A., Barkman W.E., and Babelay E.F. “Simulating the Effect of Modulated Tool-Path Chip Breaking on Surface Finish and Chip Length.” Depleted Uranium Conference 2010. Oak Ridge, TN.

[16] Smith KS, McFarland J, Tursky D, Assaid T, Barkman W, Babelay E, 2010, Simulating the Effect of Modulated Tool-Path Chip Breaking on Surface Texture and Chip Length. Y-12 Publication.

[17] McFarland, J. “Prediction and Measurement of Surface Characteristics Generated using CNC Chip Breaking Tool Paths.” UNC Charlotte. 2010.

[18] Berglind, L., Ziegert, J., 2013, Chip Breaking Parameter Selection for Constant Surface Speed Machining, ASME 2013 International Mechanical Engineering Congress and Exposition Volume 2B: Advanced Manufacturing, San Diego, CA.

[19] Berglind L, Ziegert J, Lingerfelt K, Love W. Characterization of machine axis errors for modulated tool path (MTP) machining. In Transactions of the North American Manufacturing Research Institution of SME. January ed. Vol. 42. Society of Manufacturing Engineers. 2014. p. 630-636.

[20] Berglind, L., Ziegert, J., 2015, Modulated Tool Path (MTP) Machining for Threading Applications, Procedia Manufacturing, 1: 546-555.

[21] Rubeo, M., Copenhaver, R., Landge, S., Schmitz, T., 2016, Experimental Platform for In-process Metrology during Orthogonal Turning, American Society for Precision Engineering Annual Meeting, October 23-28, Portland, OR.

[22] Copenhaver, R., Rubeo, M., Guzorek, S., Landge, S., Smith, K.S., Ziegert, J., Schmitz, T., 2017, A Fundamental Investigation of Modulated Tool Path Turning Mechanics, Procedia Manufacturing, 10: 159-170.

[23] Copenhaver, R., Schmitz, T., and Smith, S., 2018, Stability Analysis of Modulated Tool Path Turning, Annals of the CIRP, 67/1: 49-52.

[24] Toews III, H.G., Compton, W.D., and Chandrasekar, S., 1998, A Study of the Influence of Superimposed Low-frequency Modulation on the Drilling Process, Precision Engineering, 22: 1-9.

[25] Moscoso, W., Olgun, E., Compton, W.D., and Chandrasekar, S., 2005, Effect of Low-frequency Modulation on Lubrication of Chip-tool Interface in Machining. Journal of Tribology, 127: 238-244.

[26] Mann, J. B., Saldana, C., Chandrasekar, S., Compton, W.D., and Trumble, K.P., 2007, Metal Particulate Production by Modulation-assisted Machining, Scripta Materialia, 57: 909-912.

[27] Mann, J. B., Saldana, C., Moscoso, W., Compton, W.D., and Chandrasekar, S., 2009, Effects of Controlled Modulation on Interface Tribology and Deformation in Machining, Tribology Letters, 35: 221-227.
[28] Mann, J.B., Guo, Y., Saldana, C., Compton, W.D., and Chandrasekar, S., 2011, Enhancing Material Removal Processes using Modulation-assisted Machining, Tribology International, 44: 1225-1235.
[29] Mann, James B., Yang Guo, Christopher Saldana, Ho Yeung, W. Dale Compton, and Srinivasan Chandrasekar. “Modulation-Assisted Machining: A New Paradigm in Material Removal Processes.” Advanced Materials Research 223 (2011): 514–22.
[30] Guo, Y., Stalbaum, T., Mann, J., Yeung, H., and Chandrasekar, S., 2013, Modulation-assisted High Speed Machining of Compacted Graphite Iron (CGI), Journal of Manufacturing Processes, 15: 426-431.
[31] Gao, Y., Sun, R., Chen, Y., and Leopold, J., 2016, Analysis of Chip Morphology and Surface Topography in Modulation Assisted Machining, International Journal of Mechanical Sciences, 111-112: 88-100.
[32] Gao, Y., Sun, R., Chen, Y., and Leopold, J., 2016, Mechanical and Thermal Modeling of Modulation-assisted Machining, International Journal of Advanced Manufacturing Technology, 86: 2945-2959.
[33] Gao, Y., Mann, J.B., Chandrasekar, S., Sun, R., and Leopold, J., 2017, Modelling of Tool Temperature in Modulation-assisted Machining, Procedia CIRP, 58: 204-209.
[34] Gao, Y., Sun, R., and Leopold, J., 2018, An Analytical Force Model for Modulation-assisted Turning, Journal of Manufacturing Processes, 31: 712-730.
[35] Debnatha, K, and Singh, I., 2017, Low-frequency Modulation-assisted Drilling of Carbon-epoxy Composite Laminates, Journal of Manufacturing Processes, 25: 262-273.
[36] Schmitz, T. and Smith, K.S., Machining Dynamics: Frequency Response to Improved Productivity, Second Edition, Springer, New York, NY, 2019.
[37] Honeycutt, A. and Schmitz, T., 2016, A New Metric for Automated Stability Identification in Time Domain Milling Simulation, Journal of Manufacturing Science and Engineering, 138/7: 074501.
[38] Michael Gomez, Jarred Heigel, Tony Schmitz, 2018, Force modeling for hybrid manufacturing, Procedia Manufacturing, 26: 790-797,
[39] Liu, Kai, and Melkote, S.N, 2006, Effect of Plastic Side Flow on Surface Roughness in Micro-turning Process, International Journal of Machine Tools and Manufacture 46, no. 14: 1778-785.

APPENDIX

Appendix A - Additional predicted and measured surface profiles

– *OPR variation*

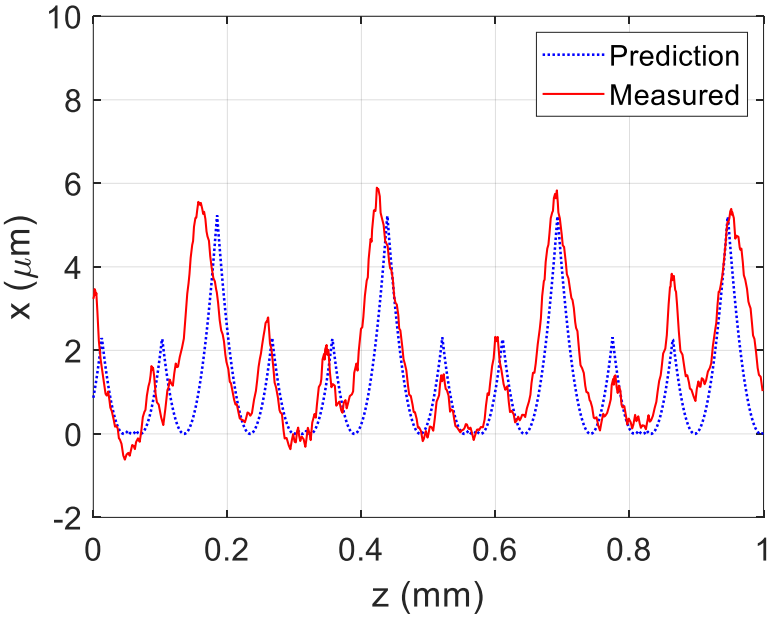

Figure A.1: Surface profile comparison for test point O31142, where the measured MTP parameters were RAF = 0.8, OPR = 0.2.

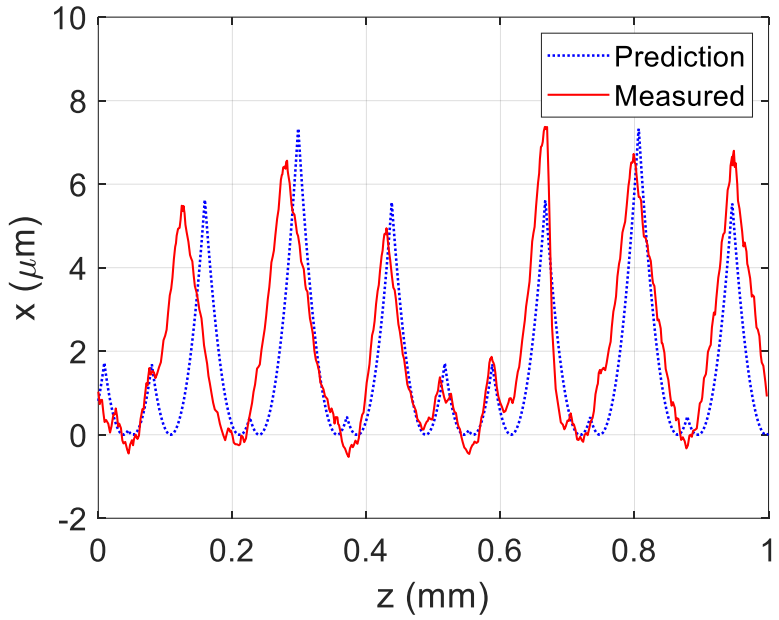

Figure A.2: Surface profile comparison for test point O31242, where the measured MTP parameters were RAF = 0.8, OPR = 0.3.

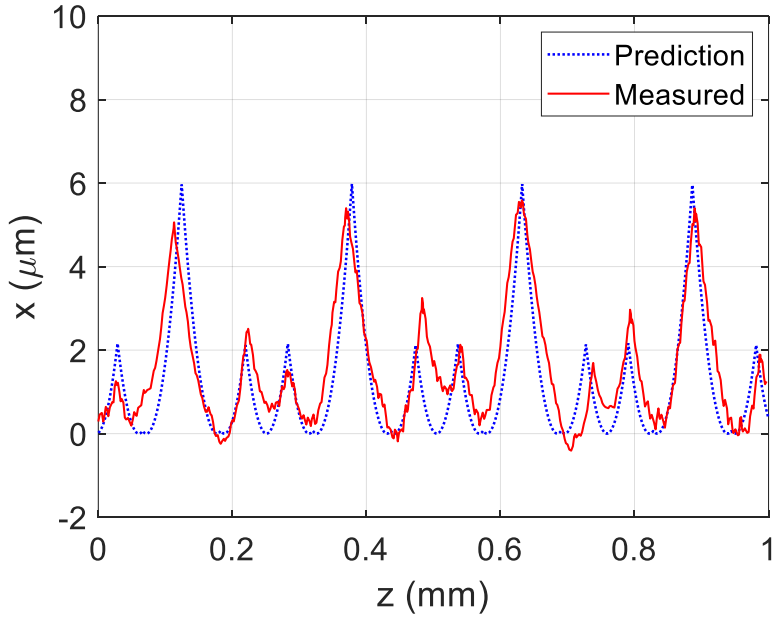

Figure A.3: Surface profile comparison for test point O31342, where the measured MTP parameters were RAF = 0.8, OPR = 0.4.

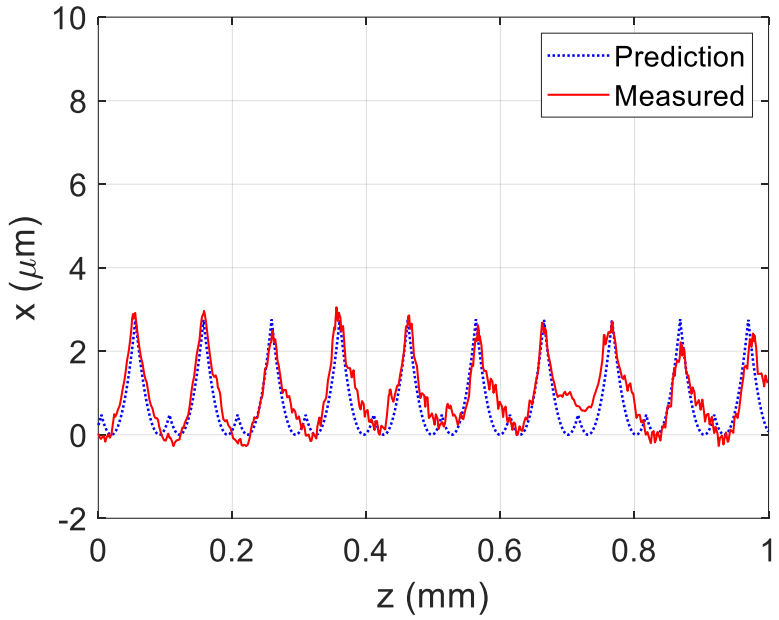

Figure A.4: Surface profile comparison for test point O31442, where the measured MTP parameters were RAF = 0.8, OPR = 0.5.

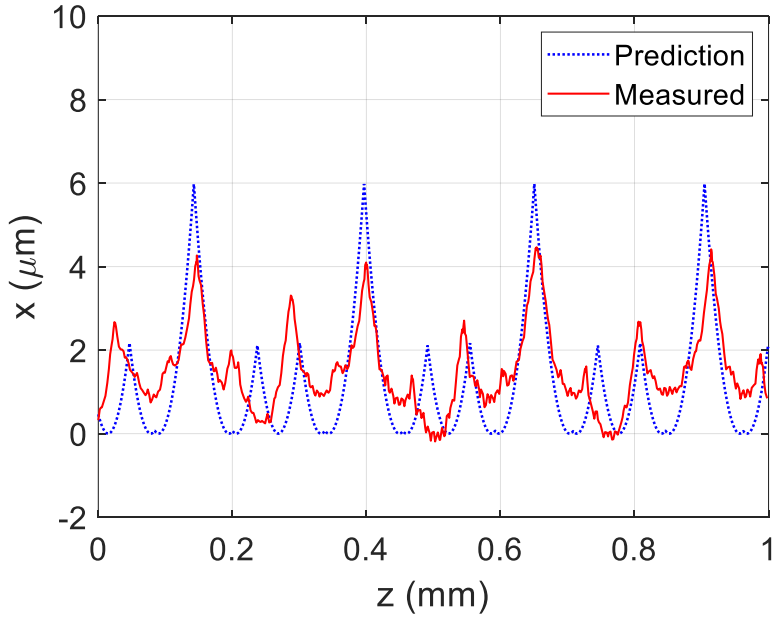

Figure A.5: Surface profile comparison for test point O31542, where the measured MTP parameters were RAF = 0.8, OPR = 0.6.

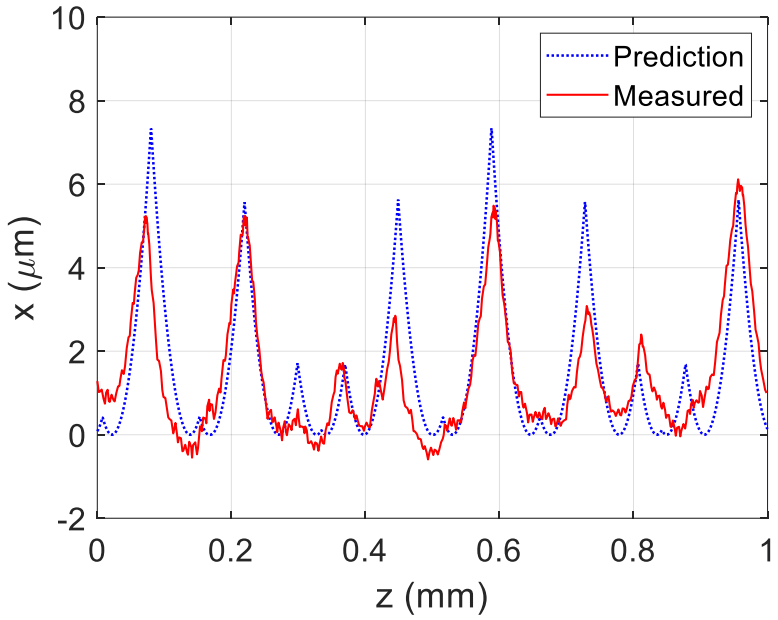

Figure A.6: Surface profile comparison for test point O31642, where the measured MTP parameters were RAF = 0.8, OPR = 0.7.

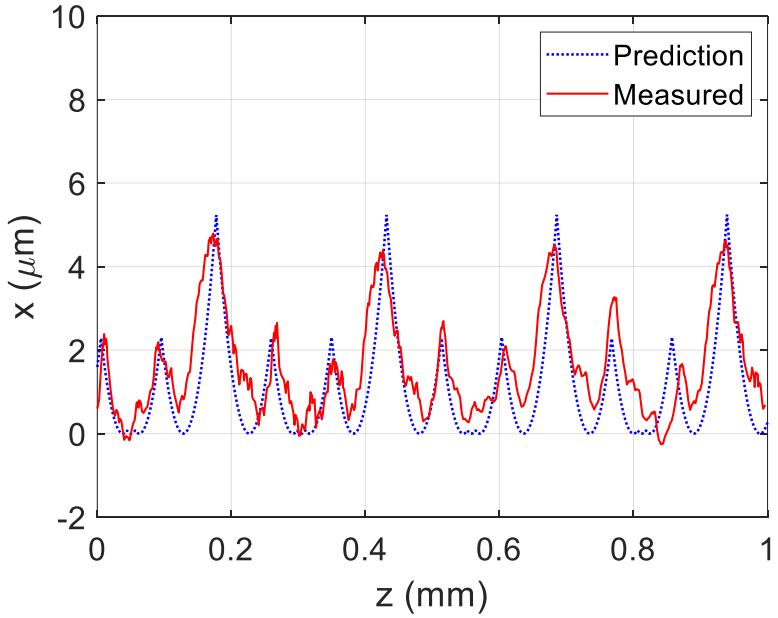

Figure A.7: Surface profile comparison for test point O31742, where the measured MTP parameters were RAF = 0.8, OPR = 0.8.

– *RAF variation*

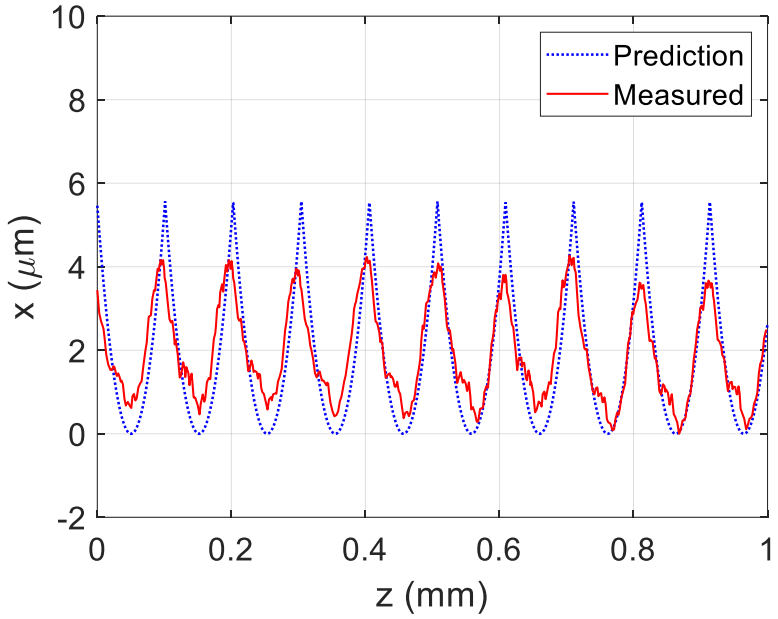

Figure A.8: Surface profile comparison for test point O31412, where the measured MTP parameters were RAF = 0.5, OPR = 0.5.

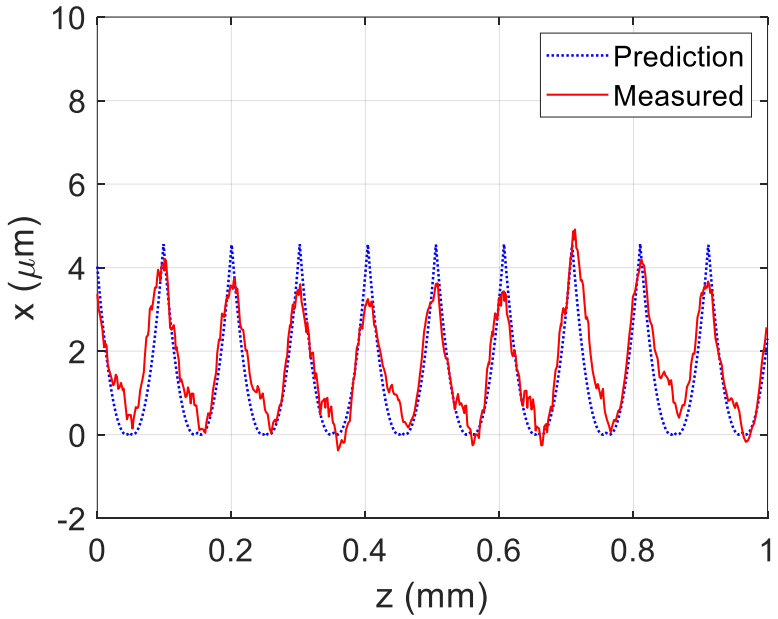

Figure A.9: Surface profile comparison for test point O31422, where the measured MTP parameters were RAF = 0.6, OPR = 0.5.

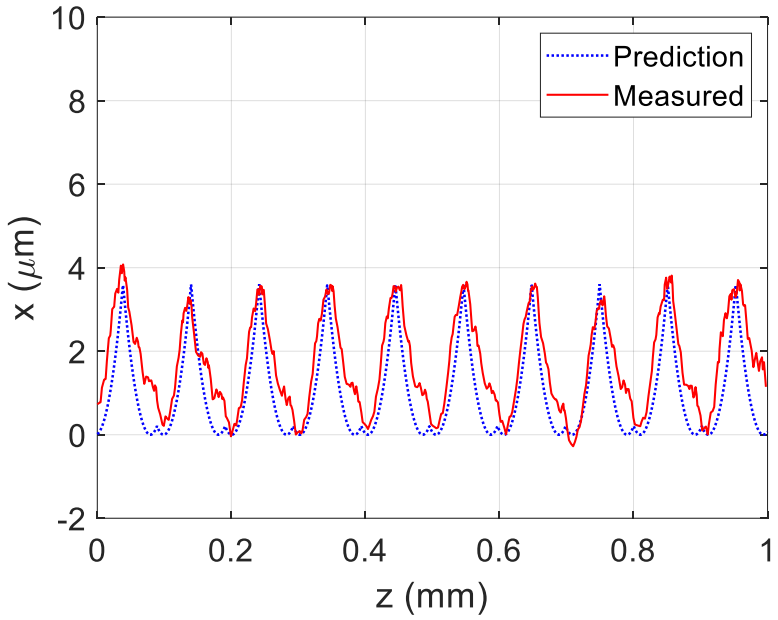

Figure A.10: Surface profile comparison for test point O31432, where the measured MTP parameters were RAF = 0.7, OPR = 0.5.

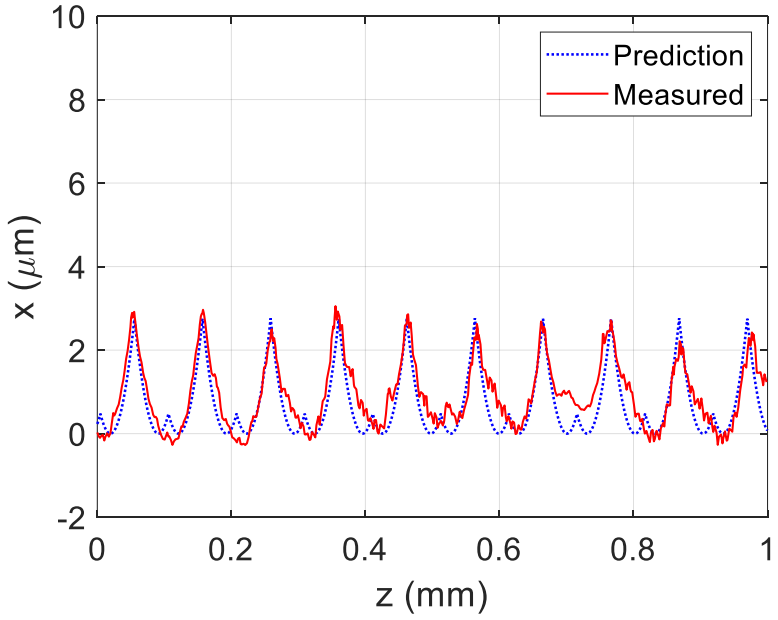

Figure A.11: Surface profile comparison for test point O31442, where the measured MTP parameters were RAF = 0.8, OPR = 0.5.

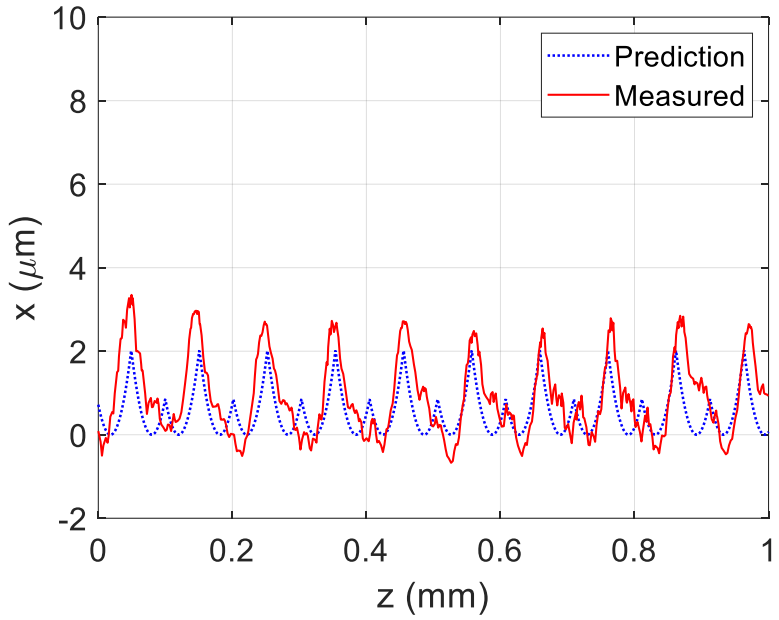

Figure A.12: Surface profile comparison for test point O31452, where the measured MTP parameters were RAF = 0.9, OPR = 0.5.

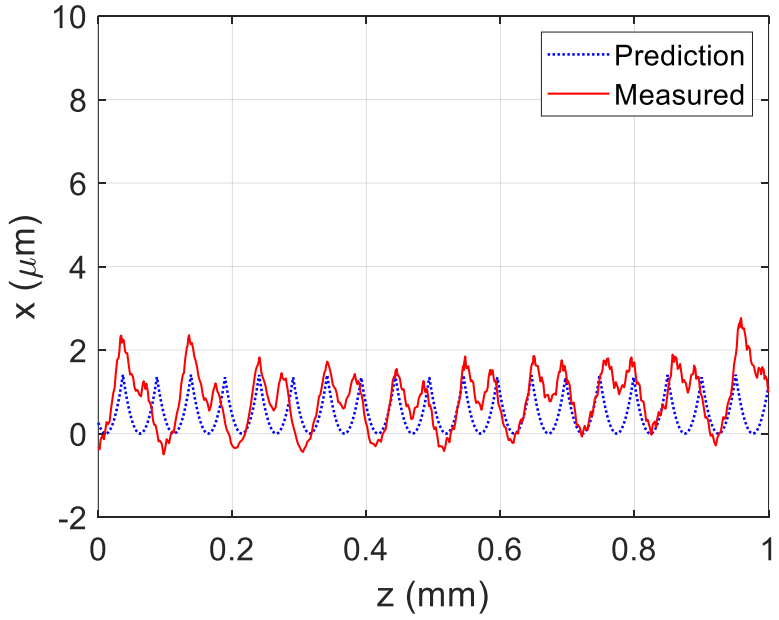

Figure A.13: Surface profile comparison for test point O31462, where the measured MTP parameters were RAF = 1.0, OPR = 0.5.

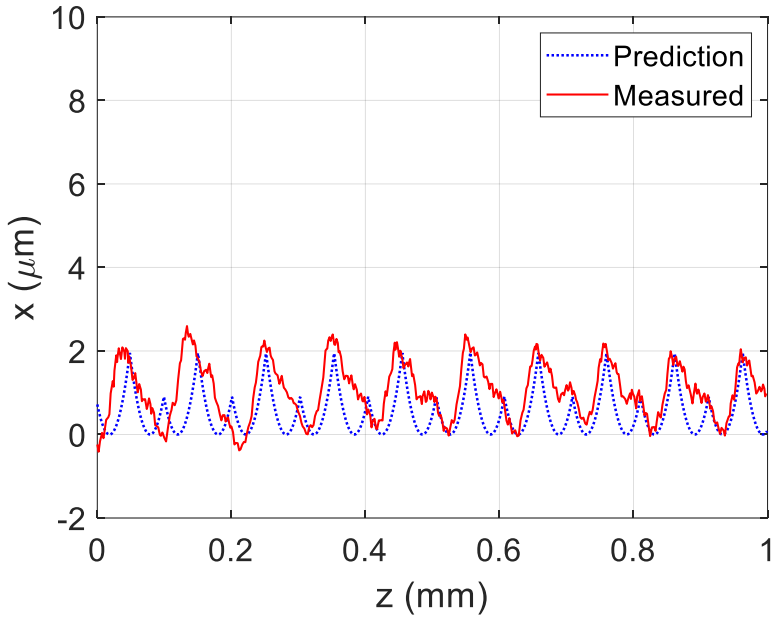

Figure A.14: Surface profile comparison for test point O31472, where the measured MTP parameters were RAF = 1.1, OPR = 0.5.

Appendix B - Appendix B – MATLAB code for single point MTP stability predictions

```
%% Header
% T. Schmitz adapted by Ryan Copenhaver
% rcopenha@uncc.edu
% 09/12/2017, adapted 05/07/2020
% mtp_simulation_single_ryan_final_version.m

% This code is used to predict the cutting force, tool
displacment, tool
% velocity, and the Poincaré map for a MTP turning
operation.

clc
close all
clear

tic

%%% Plotting Variables %%%
LW = 1; %linewidth values
FS = 16; %font size
MS = 10; %marker size

%% User input
% MTP parameters
b = 4.5e-3; % chip width, m
OPR = 0.5;
RAF = 0.8;
fr = 0.004*25.4e-3; % feed per revolution, m
time_shift = 0.015; % amount to shift the OPT sampling,
sec
rev_ammount = 75;

% Cutting parameters
omega = 556; % rpm
t_nudge = 0; % amount to shift simulated signals by,
sec

%%%%%%% Modal parameters %%%%%%%
```

```
% u1 direction (cutting, stiff direction, Y axis of
lathe)
ku1 = [8.85e8 1.73e8 2.04e8 5.97e7 5.27e9 5.15e7 3.09e8
3.73e8];    % N/m
zetau1 = [2.94 2.91 2.67 9.85 0.7 6 4 5.71]*0.01;
wnu1 = [298.5 361.3 390.5 428.9 672.8 732.1 767.5
909.1]*2*pi;   % rad/s

% u2 direction (thrust, flexibule direction, Z axis of
lathe)
ku2 = [3.43e8 5.9e7 1.73e9 4.37e8 7.5e7 8.55e8 6.49e8
2.27e8 9.69e8 3.46e8];                  % N/m
zetau2 = [2.46 3.66 2.19 2 7.46 1.2 3.12 14.21 6.18
8.24]*.01;
wnu2 = [296.0 318.2 364.5 491.3 527.6 557.7 623.8 775.1
1002.2 1375.6]*2*pi;              % rad/s

%% dynamics calculations
mu1 = ku1./(wnu1.^2);                   % kg
cu1 = 2*zetau1.*(mu1.*ku1).^0.5;        % N-s/m
u1_modes = length(ku1);                 % number of
modes in u1 direction, integer

mu2 = ku2./(wnu2.^2);                   % kg
cu2 = 2*zetau2.*(mu2.*ku2).^0.5;        % N-s/m
u2_modes = length(ku2);                 % number of
modes in u2 direction, integer

%% Setup for simulation
% Mode directions
alpha1 = 90;                            % deg
alpha2 = 90 - alpha1;
alpha1 = alpha1*pi/180;                 % rad
alpha2 = alpha2*pi/180;

fnu1 = ((ku1./mu1).^0.5)/2/pi;          % Hz
fnu2 = ((ku2./mu2).^0.5)/2/pi;          % Hz
maxfn = max([fnu1 fnu2]);
dt = 1/(100*maxfn);                     % time step
size, s
w = (omega/60)*2*pi;                    % rotating
frequency, rad/s
```

```
steps_rev = round(1/(dt*omega/60));     % number of
steps per revolution
sampling_period = round(steps_rev/OPR); % number of
samples for once per period sampling
OPR = steps_rev/sampling_period;        % reset OPR
with integer values of steps_rev and sampling_period
time_rev = 60/omega;                    % time per
revolution, s/rev

% determine the number of revolutions
% option #1 - manual
if OPR == 0 || RAF == 0
    num_rev = rev_ammount;
else
    num_rev = round(rev_ammount/OPR);                %
number of revolutions, analyze final 10 MTP periods
end

% option #2 - automated
% if OPR == 0 || RAF == 0
%     num_rev = 350;
% elseif (0 < OPR) && (OPR <= 0.15)
%     num_rev = round(150/OPR);
% elseif (0.15 < OPR) && (OPR <= 0.5)
%     num_rev = round(250/OPR);
% elseif (0.5 < OPR) && (OPR <= 1)
%     num_rev = round(500/OPR);
% elseif (1 < OPR) && (OPR <= 1.5)
%     num_rev = round(750/OPR);
% elseif (1.5 < OPR) && (OPR <= 2)
%     num_rev = round(1000/OPR);
% elseif (2 < OPR) && (OPR <= 2.5)
%     num_rev = round(1250/OPR);
% elseif (2.5 < OPR) && (OPR <= 3)
%     num_rev = round(1500/OPR);
% else
%     num_rev = round(1500/OPR);                  %
number of revolutions, analyze final 10 MTP periods
% end
total_steps = num_rev*(steps_rev + 1);  % total steps

% Define MTP motion
```

```
t = (0:total_steps-1)*dt;                                    %
time, s
z = (omega/60*fr)*t + RAF*fr*sin(w*OPR*t);                   %
feed motion, m, positive into cut
dz = (omega/60*fr) + w*OPR*RAF*fr*sin(w*OPR*t);              %
feed velocity, m/s, positive into cut

% Set zero initial conditions
u1 = 0; velu1 = 0;
u2 = 0; velu2 = 0;
dp = zeros(1, u1_modes);
p = zeros(1, u1_modes);
dq = zeros(1, u2_modes);
q = zeros(1, u2_modes);

% Predefine vectors
Force = zeros(1, total_steps);
F_thrust = zeros(1, total_steps);
F_cutting = zeros(1, total_steps);
zz = zeros(1, total_steps);     % tool motion, m,
positive out of cut
dzz = zeros(1, total_steps);    % tool velocity, m/s,
positive out of cut
ddzz = zeros(1, total_steps);    % tool acceleration,
m/s^2, positive out of cut
thick = zeros(1, total_steps);
kn_keep = zeros(1, total_steps);
kt_keep = zeros(1, total_steps);
Ks_keep = zeros(1, total_steps);
zt_t = zeros(1, total_steps);
zt_n = zeros(1, total_steps);

%% Simulation begins here
for n = (steps_rev + 1):total_steps
   zmax = z(n-steps_rev) - zz(n-steps_rev);                  %
surface from prior pass

   for cnt = 2:(ceil(n/steps_rev) - 1)
       ztest = z(n-cnt*steps_rev) - zz(n-
cnt*steps_rev);
       if ztest > zmax
```

```
            zmax = ztest;         % select maximum
surface from all previous passes
        end
    end

    h = (z(n) - zz(n-1)) - zmax;      % instantanous chip
thickness
    if h <= 0      % no cutting
        h = 0;
        F = 0;
    else

        %%% Force model - tubular workpiece %%%
        % normal (feed) dir - z axis of lathe
        kn = -3355.*(h*1e3).^(0.81) + 2520; % N/mm^2,
power 2, MTP stability testing, final

        % tangential (tool height) dir - y axis of lathe
        kt = -3491*(h*1e3).^(0.22) + 4794; % N/mm^2,
power 2, MTP stability testing, final

        kn = kn*1e6; % N/m^2
        kt = kt*1e6; % N/m^2

        Ks = (kt^2 + kn^2)^0.5;
        beta = atan(kt/kn); % rad
%         Ks = 3175e6;
%         beta = 55*pi/180;
        F = Ks*b*h;                          % N
    end
    thick(n) = h;
    kn_keep(n) = kn;
    kt_keep(n) = kt;
    Ks_keep(n) = Ks;

    Fu1 = F*cos(beta - alpha1);
    Fu2 = F*cos(beta + alpha2);
    Force(n) = F; % N
    F_thrust(n) = F*cos(beta); % N
    F_cutting(n) = F*sin(beta); % N

    % Perform Euler integrations
    u1 = 0;
```

```
    velu1 = 0;
    accelu1 = 0;
    u2 = 0;
    velu2 = 0;
    accelu2 = 0;

    % u1 direction (cutting)
    for cnt = 1:u1_modes
        ddp = (Fu1 - cu1(cnt)*dp(cnt) -
ku1(cnt)*p(cnt))/mu1(cnt);
        accelu1 = accelu1 + ddp;
        dp(cnt) = dp(cnt) + ddp*dt;
        velu1 = velu1 + dp(cnt);
        p(cnt) = p(cnt) + dp(cnt)*dt;
        u1 = u1 + p(cnt);         % m
    end

    % u2 direction (thrust)
    for cnt = 1:u2_modes
        ddq = (Fu2 - cu2(cnt)*dq(cnt) -
ku2(cnt)*q(cnt))/mu2(cnt);
        accelu2 = accelu2 + ddq;
        dq(cnt) = dq(cnt) + ddq*dt;
        velu2 = velu2 + dq(cnt);
        q(cnt) = q(cnt) + dq(cnt)*dt;
        u2 = u2 + q(cnt);         % m
    end

    zt_t(n) = u1;
    zt_n(n) = u2;

    % thrust direction
    zz(n) = u1*cos(alpha1) + u2*cos(alpha2);  % m
    dzz(n) = velu1*cos(alpha1) + velu2*cos(alpha2);  %
m/s
    ddzz(n) = accelu1*cos(alpha1) + accelu2*cos(alpha2);
% m/s^2
end  % end of simulation for loop

% remove transients
t = t((round(4*length(t)/5)):length(t));
```

```
t = t - t(1); % shifts the trimmed plot over so that
the first value is at zero
dz = dz((round(4*length(dz)/5)):length(dz));
thick =
thick((round(4*length(thick)/5)):length(thick));
kn_keep =
kn_keep((round(4*length(kn_keep)/5)):length(kn_keep));
kt_keep =
kt_keep((round(4*length(kt_keep)/5)):length(kt_keep));
Ks_keep =
Ks_keep((round(4*length(Ks_keep)/5)):length(Ks_keep));
Force =
Force((round(4*length(Force)/5)):length(Force));
F_thrust =
F_thrust((round(4*length(F_thrust)/5)):length(F_thrust)
);
F_cutting =
F_cutting((round(4*length(F_cutting)/5)):length(F_cutti
ng));
z = z((round(4*length(z)/5)):length(z));
zz = zz((round(4*length(zz)/5)):length(zz));
dzz = dzz((round(4*length(dzz)/5)):length(dzz));
ddzz = ddzz((round(4*length(ddzz)/5)):length(ddzz));
zt_t = zt_t((round(4*length(zt_t)/5)):length(zt_t));
zt_n = zt_n((round(4*length(zt_n)/5)):length(zt_n));

% ensure that that each the vector lenght is even
N = length(t);
if rem(N, 2) == 1
    t = t(1:N-1);
    dz = dz(1:N-1);
    thick = thick(1:N-1);
    Force = Force(1:N-1);
    F_thrust = F_thrust(1:N-1);
    F_cutting = F_cutting(1:N-1);
    z = z(1:N-1);
    zz = zz(1:N-1);
    dzz = dzz(1:N-1);
    ddzz = ddzz(1:N-1);
    zt_t = zt_t(1:N-1);
    zt_n = zt_n(1:N-1);
end
```

```
% sample at SR/OPR
if OPR == 0 || RAF == 0
    SR_OPR = steps_rev;
else
    SR_OPR = sampling_period;      % sample at period
that includes OPR contribution
end

delay = find(t >= time_shift);

t_s = t(delay:SR_OPR:length(t));
dz_s = dz(delay:SR_OPR:length(dz));
thick_s = thick(delay:SR_OPR:length(thick));
Force_s = Force(delay:SR_OPR:length(Force));
F_thrust_s = F_thrust(delay:SR_OPR:length(F_thrust));
F_cutting_s =
F_cutting(delay:SR_OPR:length(F_cutting));
zz_s = zz(delay:SR_OPR:length(zz));
dzz_s = dzz(delay:SR_OPR:length(dzz));
ddzz_s = ddzz(delay:SR_OPR:length(ddzz));
zt_t_s = zt_t(delay:SR_OPR:length(zt_t));
zt_n_s = zt_n(delay:SR_OPR:length(zt_n));

% zz_s = zz_s((length(zz_s)-50):length(zz_s));     %
keep final 50 MTP periods to discard initial transients

metric_disp = sum(abs(diff(zz_s)))/length(zz_s) %
metric calculation of the displacment vector

%% Plot results
% tool motion (Disp)
% figure(11)
figure(4)
hold on
a = plot(t + t_nudge, zz*1e6, 'b --', t_s + t_nudge,
zz_s*1e6, 'k o');
% a(1).Color = [0 .45 .74];
% a(1).Color = [0 0 0];
a(1).MarkerSize = 10;
a(1).LineWidth = 1.0;
a(2).MarkerSize = 10;
```

```
a(2).LineWidth = 1.5;
xlabel('t (s)')
ylabel('z_t (\mum)')
set(gca,'FontSize', FS)
ylim([-40 120]);
xlim([t(1) t(end)]);
grid on

%fft disp
N = length(zz);
fs = 1/dt;                    % sampling frequency, Hz
zz_mean = mean(zz);
zz2 = zz - zz_mean;     % remove mean prior to computing
FFT
ZZ = fft(zz2');
ZZ = ZZ(1:round(N/2+1));
ZZ = ZZ/(N/2);                % correct amplitude
ZZ(1) = zz_mean;              % replace DC value with
mean
fzz = [0:fs/N:(1-1/(2*N))*fs]';
fzz = fzz(1:round(N/2+1));             % frequency, Hz

% figure(5)
% subplot(211)
% a = plot(t + t_nudge, zz*1e6, 'b --');
% a(1).LineWidth = 1.0;
% xlabel('t (s)')
% ylabel('z_t (\mum)')
% set(gca,'FontSize', FS)
% axis([0 max(t) -150 200])
% grid on
%
% subplot(212)
% plot(fzz, abs(ZZ*1e6), 'k -')
% set(gca,'FontSize', 14)
% xlabel('f (Hz)')
% ylabel('|z_t| (\mum)')
% ylim([0 125]);
% xlim([0 2000]);
% grid on
% axes('position', [0.375 0.25 0.15 0.15]);
% box on
```

```
% index_2 = 0 < fzz & fzz < 20;
% plot(fzz(index_2), abs(ZZ(index_2)*1e6), 'k -')
% grid on
% set(gca,'FontSize', FS)

% figure(6)
% subplot(211)
% a = plot(t + t_nudge, zt_n*1e6, 'b --');
% a(1).LineWidth = 1.0;
% ylabel('zn_t (\mum)')
% set(gca,'FontSize', FS)
% axis([0 max(t) 0 15])
% grid on
%
% subplot(212)
% a = plot(t + t_nudge, zt_t*1e6, 'r --');
% a(1).LineWidth = 1.0;
% xlabel('t (s)')
% ylabel('zt_t (\mum)')
% set(gca,'FontSize', FS)
% axis([0 max(t) 0 15])
% grid on

% force
figure(12)
% subplot(211)
c = plot(t + t_nudge, F_thrust, 'b --', t_s + t_nudge,
F_thrust_s, 'k o');
% c(1).Color = [.47 .67 .19];
c(1).MarkerSize = 10;
c(1).LineWidth = 1.0;
c(2).MarkerSize = 10;
c(2).LineWidth = 1.5;
xlabel('t (s)')
ylabel('F_n (N)')
set(gca,'FontSize', FS)
ylim([-50 1500]);
xlim([t(1) t(end)]);
grid on

% subplot(212)
```

```
% c = plot(t + t_nudge, F_cutting, 'r --', t_s +
t_nudge, F_cutting_s, 'k o');
% % c(1).Color = [.47 .67 .19];
% c(1).MarkerSize = 10;
% c(1).LineWidth = 1.0;
% c(2).MarkerSize = 10;
% c(2).LineWidth = 1.5;
% xlabel('t (s)')
% ylabel('F_t (N)')
% set(gca,'FontSize', FS)
% ylim([0 500]);
% xlim([t(1) t(end)]);
% grid on

% chip thickness
% figure(13)
% plot(t, thick*1e3, 'b', t_s, thick_s*1e3, 'ro')
% set(gca,'FontSize', 14)
% xlabel('t (s)')
% ylabel('h (mm)')

% velocity from vibrometer (tool carried on cross
slide)
% figure(13)
% hold on
% b = plot(t + t_nudge, dz-dzz, 'b :', t_s + t_nudge,
dz_s-dzz_s, 'g o');
% b(1).Color = [.83 .33 .10];
% b(1).MarkerSize = MS;
% b(1).LineWidth = LW;
% b(2).MarkerSize = MS;
% b(2).LineWidth = LW+1;
% xlabel('t (s)')
% ylabel('dz_f/dt - dz_t/dt (m/s)')
% set(gca,'FontSize', FS)
% xlim([t(1) t(end)]);
% grid on

% % Calculate FFT of time domain force signal
% Force = Force(round(length(Force)/2):length(Force));
% remove initial transients
% N = length(Force);
```

```
% fs = 1/dt;                    % sampling frequency, Hz
% F_mean = mean(Force);
% Force2 = Force - F_mean;     % remove mean prior to
computing FFT
% F = fft(Force2');
% F = F(1:round(N/2+1));
% F = F/(N/2);                 % correct amplitude
% F(1) = F_mean;               % replace DC value with
mean
% f = [0:fs/N:(1-1/(2*N))*fs]';
% f = f(1:round(N/2+1));              % frequency, Hz

% %fft disp
% N = length(zz);
% zz_mean = mean(zz);
% zz2 = zz - zz_mean;    % remove mean prior to
computing FFT
% ZZ = fft(zz2');
% ZZ = ZZ(1:round(N/2+1));
% ZZ = ZZ/(N/2);                % correct amplitude
% ZZ(1) = zz_mean;              % replace DC value with
mean
% fzz = [0:fs/N:(1-1/(2*N))*fs]';
% fzz = fzz(1:round(N/2+1));             % frequency,
Hz

% figure(15)
% subplot(211)
% plot(f, abs(F))
% set(gca,'FontSize', 14)
% xlabel('f (Hz)')
% ylabel('|F| (N)')
% xlim([0 2000])
%
% subplot(212)
% plot(fzz, abs(ZZ))
% set(gca,'FontSize', 14)
% xlabel('f (Hz)')
% ylabel('|ZZ| (m)')
% xlim([0 2000])

% figure(16)
```

```
% plot(dzz, ddzz, 'g', dzz_s, ddzz_s, 'ko')
% set(gca,'FontSize', 14)
% xlabel('dz_t/dt (m/s)')
% ylabel('d^2z_t/dt^2 (m/s^2)')

figure(14)
d = plot(zz*1e6, dzz*1e3, 'g -', zz_s*1e6, dzz_s*1e3,
'ko');
% d(1).Color = [0 .45 .74];
d(1).MarkerSize = MS;
d(1).LineWidth = LW;
d(2).MarkerSize = MS;
d(2).LineWidth = LW+1;
xlabel('z_t (\mum)')
ylabel('dz_t/dt (mm/s)')
set(gca,'FontSize', FS)
xlim([-20 100]);
% ylim([-0.2 0.2]);
ylim([-200 200]);
grid on

toc
```

Appendix C - Appendix C – MATLAB code for MTP stability map predictions

```
%% Header
% T. Schmitz adapted by Ryan Copenhaver
% rcopenha@uncc.edu
% 09/13/2017, adapted 12/20/2018
% MTP_sim_multi_ryan_cluster_b_chip_width.m

% This code is used to generate the MTP stabilty map
figures resulting from
% the time domain simulation. A smaller grid set
results in longer
% simulation time.

clc
close all
clear

%% User input
% MTP parameters
RAF_vector = 0:0.05:3;
OPR_vector = 0:0.05:3;

% Cutting parameters
omega = 556; % rpm
b = 4.5e-3; % chip width, m
fr = 0.004*25.4e-3; % feed per revolution, m

% Modal parameters
% u1 direction (cutting, stiff direction)
ku1 = [8.85e8 1.73e8 2.04e8 5.97e7 5.27e9 5.15e7 3.09e8
3.73e8]    % N/m
zetau1 = [2.94 2.91 2.67 9.85 0.7 6 4 5.71]*0.01;
wnu1 = [298.5 361.3 390.5 428.9 672.8 732.1 767.5
909.1]*2*pi;   % rad/s

mu1 = ku1./(wnu1.^2)                    % kg
cu1 = 2*zetau1.*(mu1.*ku1).^0.5         % N-s/m
u1_modes = length(ku1);                  % number of
modes in u1 direction, integer
```

```
% u2 direction (thrust, flexibule direction)
ku2 = [3.43e8 7.06e7 1.73e9 4.37e8 4.66e7 8.55e8 6.49e8
2.27e8 9.69e8 3.46e8]                    % N/m
zetau2 = [2.46 3.66 2.19 2 7.46 1.2 3.12 14.21 6.18
8.24]*.01;
wnu2 = [296.0 318.2 364.5 491.3 527.6 557.7 623.8 775.1
1002.2 1375.6]*2*pi;                % rad/s

mu2 = ku2./(wnu2.^2)                     % kg
cu2 = 2*zetau2.*(mu2.*ku2).^0.5          % N-s/m
u2_modes = length(ku2);                   % number of
modes in u2 direction, integer

%% Setup for simulation
% Mode directions
alpha1 = 90;                              % deg
alpha2 = 90 - alpha1;
alpha1 = alpha1*pi/180;                   % rad
alpha2 = alpha2*pi/180;

phi = 2*pi*(OPR_vector - floor(OPR_vector));     % rad
RAF_lim = 1./(2*sin(phi/2));

% Simulation inputs
fnu1 = ((ku1./mu1).^0.5)/2/pi;           % Hz
fnu2 = ((ku2./mu2).^0.5)/2/pi;           % Hz
maxfn = max([fnu1 fnu2]);

rows = length(OPR_vector);
cols = length(RAF_vector);
metric = zeros(rows, cols);

%% Simulation begins here
for cnt1 = 1:rows
    cnt1

    OPR = OPR_vector(cnt1);

    for cnt2 = 1:cols
        RAF = RAF_vector(cnt2);

        dt = 1/(50*maxfn);                       % s
```

```
        w = (omega/60)*2*pi;                        %
rotating frequency, rad/s
        steps_rev = round(1/(dt*omega/60));         %
number of steps per revolution
        time_rev = 60/omega;                        % time
per revolution, s/rev
        dt = time_rev/steps_rev;                    % time
step size, s
        sampling_period = round(steps_rev/OPR); %
number of samples for once per period sampling
        OPR = steps_rev/sampling_period;            % reset
OPR with integer values of steps_rev and
sampling_period

        if OPR == 0 || RAF == 0
            num_rev = 350;
        elseif (0 < OPR) && (OPR <= 0.15)
            num_rev = round(150/OPR);
        elseif (0.15 < OPR) && (OPR <= 0.5)
            num_rev = round(250/OPR);
        elseif (0.5 < OPR) && (OPR <= 1)
            num_rev = round(500/OPR);
        elseif (1 < OPR) && (OPR <= 1.5)
            num_rev = round(750/OPR);
        elseif (1.5 < OPR) && (OPR <= 2)
            num_rev = round(1000/OPR);
        elseif (2 < OPR) && (OPR <= 2.5)
            num_rev = round(1250/OPR);
        elseif (2.5 < OPR) && (OPR <= 3)
            num_rev = round(1500/OPR);
        else
            num_rev = round(1500/OPR);                  %
number of revolutions, analyze final 10 MTP periods
        end
        total_steps = num_rev*(steps_rev + 1);  % total
steps

        % Define MTP motion
        t = (0:total_steps-1)*dt;                   % time,
s
        z = (omega/60*fr)*t + RAF*fr*sin(w*OPR*t);  %
feed motion, m
```

```
        % Set zero initial conditions
        u1 = 0; velu1 = 0;
        u2 = 0; velu2 = 0;
        dp = zeros(1, u1_modes);
        p = zeros(1, u1_modes);
        dq = zeros(1, u2_modes);
        q = zeros(1, u2_modes);

        % Predefine vectors
        Force = zeros(1, total_steps);
        zz = zeros(1, total_steps);
        thick = zeros(1, total_steps);

        % Simulation begins here
        for n = (steps_rev + 1):total_steps
            zmax = z(n-steps_rev) - zz(n-steps_rev);
% surface from prior revolution

            for cnt = 2:(ceil(n/steps_rev) - 1)     %
find maximum from all previous revolutions
                ztest = z(n-cnt*steps_rev) - zz(n-
cnt*steps_rev);
                if ztest > zmax
                    zmax = ztest;
                end
            end

            h = (z(n) - zz(n-1)) - zmax;     %
instantanous chip thickness
            if h <= 0     % no cutting
                h = 0;
                F = 0;
            else
                % Force model for non-zero nose radius
is included for steel
                % tubular workpiece
                kn = 1120*(h*1e3)^(-0.175);
                kn = kn*1e6;                        %
N/m^2

                kt = 1800*(h*1e3)^(-0.15);
```

```
                kt = kt*1e6;
                Ks = (kt^2 + kn^2)^0.5;
                beta = atan(kt/kn);                 % rad
                F = Ks*b*h;                         % N
            end
            thick(n) = h;

            Fu1 = F*cos(beta - alpha1);
            Fu2 = F*cos(beta + alpha2);
            Force(n) = F;                   % N

            % Perform Euler integrations
            u1 = 0;
            velu1 = 0;
            u2 = 0;
            velu2 = 0;

            % u1 direction (tangential)
            for cnt = 1:u1_modes
                ddp = (Fu1 - cu1(cnt)*dp(cnt) -
ku1(cnt)*p(cnt))/mu1(cnt);
                dp(cnt) = dp(cnt) + ddp*dt;
                velu1 = velu1 + dp(cnt);
                p(cnt) = p(cnt) + dp(cnt)*dt;
                u1 = u1 + p(cnt);           % m
            end

            % u2 direction (axial)
            for cnt = 1:u2_modes
                ddq = (Fu2 - cu2(cnt)*dq(cnt) -
ku2(cnt)*q(cnt))/mu2(cnt);
                dq(cnt) = dq(cnt) + ddq*dt;
                velu2 = velu2 + dq(cnt);
                q(cnt) = q(cnt) + dq(cnt)*dt;
                u2 = u2 + q(cnt);           % m
            end

            % Axial direction
            zz(n) = u1*cos(alpha1) + u2*cos(alpha2);  %
m
        end  % end of simulation for loop
```

```
        % sample at SR/OPR
        delay = steps_rev + 1;
        if OPR == 0 || RAF == 0
            SR_OPR = steps_rev;
        else
            SR_OPR = sampling_period;      % sample at
period that includes OPR contribution
        end
        zz_s = zz(delay:SR_OPR:length(zz));
        zz_s = zz_s((length(zz_s)-100):length(zz_s)); %
keep final 100 MTP periods to discard initial
transients
        metric(cnt1, cnt2) =
sum(abs(diff(zz_s*1e6)))/length(zz_s);   % micrometers
    end
end

%% Save data
save b_4_5_data phi RAF_lim RAF_vector OPR_vector
metric
```

Appendix D - Appendix D – MATLAB code for MTP stability map predictions

```
%% Header
% T. Schmitz adapted by Ryan Copenhaver
% rcopenh1@vols.utk.edu
% 5/16/20, adapted 10/08/2020
% MTP_TDS_Surface_Finish_Analysis.m

% Program predicts the surface finish from the time domain simulation for
% constant feed and MTP feed turning. This code is intended to help find
% the sensitivity of the predicted surface finish to the input parameters
% (OPR, RAF, and feed rate)

clear
close all
clc

tic

%%% Plotting Variables %%%
LW = 1; %linewidth values
FS = 16; %font size
MS = 10; %marker size

%% User input
%%% number of revolutions to disregard to remove transients %%%
n_rev_remove = 25; % number of revolutions to remove from the n_vector

% %%% OPR = 0.5, RAF = 0.8, f = 0.002 in/rev, nominal sensitivity analysis
b = (0.005)*25.4*1e-3; % chip width, m
omega = 1056; % rpm
OPR = 0.5; % OPR value the sensitivity analysis starts at
```

```
RAF = 0.8; % RAF value the snesitivity analisi starts
at
fr = (0.002)*25.4*1e-3; % orignial feed per revolution,
m
time_shift = 0; % amount to shift the OPT sampling, sec
t_nudge = 0; % amount to shift simulated signals by,
sec
part_raduis = (2.89)*(1/2)*25.4*1e-3; % workpiece
radius, m
rev_ammount = 75 + n_rev_remove;
radi_placment = 0.5; % smaple surface at X percent
through a revolution
save_name = 'example_code';
nr = (0.0092)*25.4*1e-3; % tool nose radius, m


%% Save names, locations, and plot bounds
%%% manual name input
figs_name = sprintf('%s_figs', save_name); % Saved
figures name
data_name = sprintf('%s', save_name); % Saved data name
% % orignial data location
% data_compare_save_location =
('C:\Users\rcopi\Desktop\Y12_Surface_finish_project\OD_
turn_test\AL_6061_Testing\TDS_Sensitivity_Analysis\Save
d_TDS_Profiles_Data_and_Figs'); % where to save the TDS
and meas profile comparision

% mess arround data location
data_compare_save_location =
('C:\Users\rcopi\Desktop\Y12_Surface_finish_project\OD_
turn_test\AL_6061_Testing\TDS_Sensitivity_Analysis\Save
d_TDS_Profiles_Data_and_Figs\OPR_0.5_RAF_0.8_fr_0.002_i
pr_sensitvity_analysis'); % where to save the TDS and
meas profile comparision
plot_axes_bounds = [1 4 -1 16]; % the plot axes

%% Tool modal parameters
% Modal parameters are in the plane of the cut (feed
direction, lathe YZ plane)
%%%%%%%%%%%%%% u1 direction (cutting direction, lathe Y
axis) %%%%%%%%%%%%%%
```

```
% 35 deg neutral rake tool - Machining at UTK
wnu1_tool = [3974 4309 5653 5919 6399 7810]*2*pi; %
rad/s
ku1_tool = [6.77E+07 1.53E+08 6.38E+07 1.24E+08
6.13E+07 1.71E+08]; % N/m
zetau1_tool = [11.43 5.94 4.08 2.08 2.61 3.09]/100;

%%%%%%%%%%%%%%% u2 direction (feed direction, lathe Z
axis) %%%%%%%%%%%%%%%
% 35 deg neutral rake tool - Machining at UTK
wnu2_tool = [377 2715 4313 4985 5904 6409]*2*pi; %
rad/s
ku2_tool = [1.75E+08 1.37E+08 5.79E+07 6.05E+07
3.87E+07 1.12E+08]; % N/m
zetau2_tool = [2.17 3.19 4.59 5.6 3.09 1.67]/100;

%%%%%%%%%%%%%%% u3 direction (radial direction of
cylindrical part, lathe X axis) %%%%%%%%%%%%%%%
% 35 deg neutral rake tool - Machining at UTK
wnu3_tool = [60 114 146 181 266 329 518 558 1328
2696]*2*pi; % rad/s
ku3_tool = [8.07E+08 2.47E+09 7.20E+07 4.46E+08
1.87E+08 1.89E+09 2.67E+08 1.40E+09 3.62E+09 6.35E+08];
% N/m
zetau3_tool = [3.42 3.48 4 8.57 6.43 4.08 7.92 3.66
2.30 3.96]/100;

%% Workpiece modal parameters
%%%%%%%%%%%%%% u1 direction (cutting direction, lathe Y
axis) %%%%%%%%%%%%%%
% Free end, 3in dia AL workpiece
wnu1_work = [208 315 364 397 573 864 888 994 1328
1356]*2*pi; % rad/s
ku1_work = [1.47371E+008 3.43448E+007 4.60968E+008
2.23444E+008 2.73540E+008 1.41565E+008 1.39039E+009
2.16325E+008 1.80110E+008 2.10711E+008]; % N/m
zetau1_work = [4.08 3.09 3.15 2.08 7.17 3.66 0.8 9.37
1.87 0.71]/100;

%%%%%%%%%%%%%%% u2 direction (feed direction, lathe Z
axis) %%%%%%%%%%%%%%%
% Free end, 3in dia AL workpiece
```

```
wnu2_work = [206 255 291 314 395 1752 2781 3302
4538]*2*pi; % rad/s
ku2_work = [3.17E+08 4.56E+09 1.03E+09 4.15E+08
4.27E+09 1.27E+09 3.73E+09 1.65E+09 6.27E+09]; % N/m
zetau2_work = [3.69 2.46 4.04 3.41 1.37 2.12 0.87 1.56
1.92]/100;

%%%%%%%%%%%%%%% u3 direction (radial direction of
cylindrical part, lathe X axis) %%%%%%%%%%%%%%%
% Free end, 3in dia AL workpiece
wnu3_work = [313 356 441 604 640 781 968 1329
1372]*2*pi; % rad/s
ku3_work = [6.76923E+007 2.96057E+008  2.74621E+008
3.52679E+008 5.98583E+008 6.04403E+008 1.15870E+008
1.34932E+008 2.81901E+008]; % N/m
zetau3_work = [1.94 3.09 4.33 1.37 1.43 2.91 4.97 2.19
1.01]/100;

%% Combine modal parameters
% combine tool and workpiece dynamics
wnu1 = [wnu1_tool wnu1_work]; % [rad/s]
ku1 = [ku1_tool ku1_work]; % [N/m]
zetau1 = [zetau1_tool zetau1_work];

wnu2 = [wnu2_tool wnu2_work]; % [rad/s]
ku2 = [ku2_tool ku2_work]; % [N/m]
zetau2 = [zetau2_tool zetau2_work];

wnu3 = [wnu3_tool wnu3_work]; % [rad/s]
ku3 = [ku3_tool ku3_work]; % [N/m]
zetau3 = [zetau3_tool zetau3_work];

% calculate the modal parameters
mu1 = ku1./(wnu1.^2); % kg
cu1 = 2*zetau1.*(mu1.*ku1).^0.5; % N-s/m
u1_modes = length(ku1); % number of modes in u1
direction, integer

mu2 = ku2./(wnu2.^2); % kg
cu2 = 2*zetau2.*(mu2.*ku2).^0.5; % N-s/m
u2_modes = length(ku2); % number of modes in u2
direction, integer
```

```
mu3 = ku3./(wnu3.^2); % kg
cu3 = 2*zetau3.*(mu3.*ku3).^0.5; % N-s/m
u3_modes = length(ku3); % number of modes in u3
direction, integer

%% Setup of simulation
% Cutting model directions (lathe, YZ plane)
alpha1 = 90;
alpha2 = 90 - alpha1;
alpha1 = alpha1*pi/180;
alpha2 = alpha2*pi/180;

fnu1 = ((ku1./mu1).^0.5)/2/pi; % Hz
fnu2 = ((ku2./mu2).^0.5)/2/pi; % Hz
fnu3 = ((ku3./mu3).^0.5)/2/pi; % Hz
maxfn = max([fnu1 fnu2 fnu3]); % find max frequency
present, Hz

dt = 1/(100*maxfn);                         % time step
size, s
w = (omega/60)*2*pi;                        % rotating
frequency, rad/s
steps_rev = round(1/(dt*omega/60));         % number of
steps per revolution
sampling_period = round(steps_rev/OPR); % number of
samples for once per period sampling
OPR = steps_rev/sampling_period;            % reset OPR
with integer values of steps_rev and sampling_period
time_rev = 60/omega;                        % time per
revolution, s/rev
dphi = dt*omega*(2*pi)/60;                  % angular step
size, rad

% determine the number of revolutions
% option #1 - manual
if OPR == 0 || RAF == 0
    num_rev = rev_ammount;
else
    num_rev = round(rev_ammount/OPR); % number of
revolutions
end
```

```
% option #2 - automated
% if OPR == 0 || RAF == 0
%     num_rev = 350;
% elseif (0 < OPR) && (OPR <= 0.15)
%     num_rev = round(150/OPR);
% elseif (0.15 < OPR) && (OPR <= 0.5)
%     num_rev = round(250/OPR);
% elseif (0.5 < OPR) && (OPR <= 1)
%     num_rev = round(500/OPR);
% elseif (1 < OPR) && (OPR <= 1.5)
%     num_rev = round(750/OPR);
% elseif (1.5 < OPR) && (OPR <= 2)
%     num_rev = round(1000/OPR);
% elseif (2 < OPR) && (OPR <= 2.5)
%     num_rev = round(1250/OPR);
% elseif (2.5 < OPR) && (OPR <= 3)
%     num_rev = round(1500/OPR);
% else
%     num_rev = round(1500/OPR); % number of
revolutions
% end
total_steps = num_rev*(steps_rev + 1);  % total steps

% Define MTP motion
t = (0:total_steps-1)*dt; % time, s
z = (omega/60*fr)*t + RAF*fr*sin(w*OPR*t); % feed
motion, m, positive into cut in feed direction (lathe
Z)
dz = (omega/60*fr) + w*OPR*RAF*fr*sin(w*OPR*t); % feed
velocity, m/s, positive into cut

% Set zero initial conditions
u1 = 0; velu1 = 0;
u2 = 0; velu2 = 0;
u3 = 0; velu3 = 0;
dp = zeros(1, u1_modes);
p = zeros(1, u1_modes);
dq = zeros(1, u2_modes);
q = zeros(1, u2_modes);
dr = zeros(1, u3_modes);
r = zeros(1, u3_modes);
```

```
% Predefine vectors
Force = zeros(1, total_steps);
F_thrust = zeros(1, total_steps);
zz = zeros(1, total_steps); % tool motion, m, positive
out of cut in surface normal direction
dzz = zeros(1, total_steps); % tool velocity, m/s,
positive out of cut
ddzz = zeros(1, total_steps); % tool acceleration,
m/s^2, positive out of cut
thick = zeros(1, total_steps); % chip thickness, m
radial = zeros(1, total_steps); % tool motion, m,
positive away from part in radial direction

%% Simulation begins here
for n = (steps_rev + 1):total_steps
   zmax = z(n - steps_rev) - zz(n - steps_rev); %
surface from prior pass

   for cnt = 2:(ceil(n/steps_rev) - 1)
       ztest = z(n - cnt*steps_rev) - zz(n -
cnt*steps_rev);
       if ztest > zmax
           zmax = ztest; % select maximum surface from
all previous passes
       end
   end

   h = (z(n) - zz(n-1)) - zmax; % instantanous chip
thickness in plane of cut, surface normal direction
(lathe YZ)

   % Force model for non-zero nose radius is included
for steel
   %%% Force model - tubular workpiece %%%
   % normal (feed) dir - Z axis of lathe
   kn = 39.07*(h*1e3).^(-0.97) + 111.84; % N/mm^2,
linear polynomial, 6061-T6 AL, neutral rake, finish
cut, surface normal direction (lathe Z)
   kn = kn*1e6; % N/m^2

   % tangential (cutting) dir - Y axis of lathe
```

```
    kt = 131.06*(h*1e3).^(-0.89) + 702.10; % N/mm^2,
linear polynomial, 6061-T6 AL, neutral rake, finish
cut, cutting speed direction (lathe Y)
    kt = kt*1e6; % N/m^2

    % radial (raduis) dir - X axis if lathe
    kr = 124.08*(h*1e3).^(-0.99) + 564.40; % N/mm^2,
linear polynomial, 6061-T6 AL, neutral rake, finish
cut, radial direction (lathe X)
    kr = kr*1e6; % N/m^2

    % CFC and cutting force angle
    Ks = (kt^2 + kn^2)^0.5; % N/m^2
    beta = atan(kt/kn); % rad

    if h < 0 % no cutting
        h = 0;
    end
    thick(n) = h;

    F = Ks*(b - u3)*h; % reduce chip width by u3,
vibration in radial direction
    Fu1 = F*cos(beta - alpha1);
    Fu2 = F*cos(beta + alpha2);
    Force(n) = F; % N
    F_thrust(n) = F*cos(beta);
    Fu3 = kr*(b - u3)*h;

    % Perform Euler integrations
    u1 = 0;
    velu1 = 0;
    accelu1 = 0;
    u2 = 0;
    velu2 = 0;
    accelu2 = 0;
    u3 = 0;

    % u1 direction (cutting)
    for cnt = 1:u1_modes
        ddp = (Fu1 - cu1(cnt)*dp(cnt) -
ku1(cnt)*p(cnt))/mu1(cnt);
        accelu1 = accelu1 + ddp;
```

```
        dp(cnt) = dp(cnt) + ddp*dt;
        velu1 = velu1 + dp(cnt);
        p(cnt) = p(cnt) + dp(cnt)*dt;
        u1 = u1 + p(cnt);         % m
    end

    % u2 direction (surface normal in plane of cut)
    for cnt = 1:u2_modes
        ddq = (Fu2 - cu2(cnt)*dq(cnt) -
ku2(cnt)*q(cnt))/mu2(cnt);
        accelu2 = accelu2 + ddq;
        dq(cnt) = dq(cnt) + ddq*dt;
        velu2 = velu2 + dq(cnt);
        q(cnt) = q(cnt) + dq(cnt)*dt;
        u2 = u2 + q(cnt);         % m
    end

    % u3 direction (radial direction)
    for cnt = 1:u3_modes
        ddr = (Fu3 - cu3(cnt)*dr(cnt) -
ku3(cnt)*r(cnt))/mu3(cnt);
        dr(cnt) = dr(cnt) + ddr*dt;
        r(cnt) = r(cnt) + dr(cnt)*dt;
        u3 = u3 + r(cnt);         % m
    end

    % surface normal direction in plane of cut
    zz(n) = u1*cos(alpha1) + u2*cos(alpha2);  % m
    dzz(n) = velu1*cos(alpha1) + velu2*cos(alpha2);  %
m/s
    ddzz(n) = accelu1*cos(alpha1) + accelu2*cos(alpha2);
% m/s^2
    radial(n) = u3;  % m
end  % end of simulation for loop

%% Surface finish calculations
% total motion is difference of z (positive into cut)
and zz (positive out
% of cut)
z_total = z - zz;
```

```
surface_dist = (0:steps_rev-1)*dphi*part_raduis;
% surface distance, m

z_parse = z_total(1:steps_rev);
n_parse = radial(1:steps_rev);
figure('Name','z-parse, MTP tool motion
location','NumberTitle','on');
plot(surface_dist*1e3, z_total(1:steps_rev)*1e3, 'b')
set(gca,'FontSize', FS)
xlabel('d (mm)')
ylabel('z_f (mm)')
xlim([0 2*pi*part_raduis*1e3])
grid on
hold on
zoom on
for cnt = 2:num_rev
    z_parse = [z_parse; z_total(((cnt-1)*steps_rev +
1):(cnt*steps_rev))];
    if rem(cnt, 2) == 1
        plot(surface_dist*1e3, z_total(((cnt-
1)*steps_rev + 1):(cnt*steps_rev))*1e3, 'b')
    else
        plot(surface_dist*1e3, z_total(((cnt-
1)*steps_rev + 1):(cnt*steps_rev))*1e3, 'r')
    end
    n_parse = [n_parse; radial(((cnt-1)*steps_rev +
1):(cnt*steps_rev))];
end

test_point = round(steps_rev*radi_placment); % where
the noise radi are placed along the diameter of the
workpeice
z_total_test = z_parse(:, test_point);
n_total_test = n_parse(:, test_point);
surface_dist_test = ones(num_rev,
1)*surface_dist(test_point);
plot(surface_dist_test*1e3, z_total_test*1e3, 'mo')

% set nr using feed since previous revolution
fr_eff = abs(diff(z_total_test));
fr_eff = [fr; fr_eff];
```

```
% without side flow
z_temp = linspace(z_total_test(1)-0.5*nr,
z_total_test(1)+0.5*nr, 25000);
n_temp = -((nr^2 - (z_temp -
z_total_test(1)).^2).^0.5);

% % with side flow - AL, b = 0.107mm, 35 deg insert
cutting tests
% nr = zeros(1, num_rev);
% for cnt = 1:num_rev
%     if fr_eff(cnt) < 0.0508e-3 % lower feed rate
"hard stop"
%         nr(cnt) = (0.2337)*(1e-3);
%     elseif fr_eff(cnt) > 0.1003e-3 % upper upper
"hard stop"
%         nr(cnt) = (0.397)*(1e-3);
%     else
%         nr(cnt) = (3.299*(fr_eff(cnt)*1e3) +
0.0661)*1e-3; % linear fit in-between the "hard stops"
%     end
% end
% z_temp = linspace(z_total_test(1)-0.5*nr(1),
z_total_test(1)+0.5*nr(1), 25000);
% n_temp = -((nr(1)^2 - (z_temp -
z_total_test(1)).^2).^0.5) + n_total_test(1);

figure('Name','n-parse, tool nose radius
superposition','NumberTitle','on');
plot(z_temp*1e3, n_temp*1e3 - min(n_temp)*1e3, 'b')
set(gca,'FontSize', FS)
xlabel('z (mm)')
ylabel('n (mm)')
grid on
hold on
zoom on
%ylim([(-nr*1e3) (-nr*1e3 + 1e-3)])

% without side flow
z_interp = linspace(z_total_test(1)-0.5*nr,
z_total_test(num_rev)+0.5*nr, 1e5);
n_interp = interp1(z_temp, n_temp, z_interp);
n_vector = [n_interp];
```

```
for cnt = 2:num_rev
    z_temp = linspace(z_total_test(cnt)-0.5*nr,
z_total_test(cnt)+0.5*nr, 25e3);
    n_temp = -((nr^2 - (z_temp -
z_total_test(cnt)).^2).^0.5);
    if rem(cnt, 2) == 1
        plot(z_temp*1e3, n_temp*1e3 - min(n_temp)*1e3,
'b')
    else
        plot(z_temp*1e3, n_temp*1e3 - min(n_temp)*1e3,
'r')
    end
    n_interp = interp1(z_temp, n_temp, z_interp);
    n_vector = [n_vector; n_interp];
end

% % with side flow
% z_interp = linspace(z_total_test(1)-0.5*nr(1),
z_total_test(num_rev)+0.5*nr(1), 1e5);
% n_interp = interp1(z_temp, n_temp, z_interp);
% n_vector = [n_interp];
% for cnt = 2:num_rev
%     z_temp = linspace(z_total_test(cnt)-0.5*nr(cnt),
z_total_test(cnt)+0.5*nr(cnt), 25e3);
%     n_temp = -((nr(cnt)^2 - (z_temp -
z_total_test(cnt)).^2).^0.5) + n_total_test(cnt);
%     if rem(cnt, 2) == 1
%         plot(z_temp*1e3, n_temp*1e3, 'b')
%     else
%         plot(z_temp*1e3, n_temp*1e3, 'r')
%     end
%     n_interp = interp1(z_temp, n_temp, z_interp);
%     n_vector = [n_vector; n_interp];
% end

n_final = min(n_vector);
[peaks, index] = findpeaks(-n_final);
n_final =
n_final(index(n_rev_remove*2):index(length(index))); %
Remove transients
```

```
z_interp =
z_interp(index(n_rev_remove*2):index(length(index))); %
Remove transients
n_final = n_final - mean(n_final);

% set the bottom of the plot to zero
n_final_min = min(n_final(1:end));
n_final = n_final - n_final_min;

% move the parsed vector over to start of the time
vector
z_interp = z_interp - z_interp(1);

% transpose and rename vectors
x_tds_m = n_final';
z_tds_m = z_interp';
x_tds_in = (n_final')*((1e3)/(25.4));
z_tds_in = (z_interp')*((1e3)/(25.4));

%% Plot profiles
fig(1) = figure('Name','Inch TDS Surface
profile','NumberTitle','on');
plot(z_tds_m*(1e3)/25.4, x_tds_m*(1e9)/25.4,
'b:','LineWidth', LW+0.25)
legend('Predict')
xlabel('z (in)')
ylabel('x (\muin)')
% xlim([0, max(z_trace_length_m)*(1e3)/25.4])
% axis(plot_axes_bounds/25.4)
set(gca,'FontSize', FS)
grid on

fig(2) = figure('Name','Metric TDS Surface
profile','NumberTitle','on');
plot(z_tds_m*(1e3), x_tds_m*(1e6), 'b:', 'LineWidth',
LW+0.25)
legend('Predict')
xlabel('z (mm)')
ylabel('x (\mum)')
% xlim([0, max(z_trace_length_m)*(1e3)])
% axis(plot_axes_bounds)
set(gca,'FontSize', FS)
```

```
grid on

% Calculate Ra
Ra_tds_micron = sum(abs(x_tds_m)*1e6)/(length(x_tds_m))
Ra_tds_micro_in = Ra_tds_micron*25.4
sprintf('Ra_tds_micron_%s', Ra_tds_micron);
sprintf('Ra_tds_micro_in_%s', Ra_tds_micro_in);

% cd(data_compare_save_location); %Imports the saving
directory
% % savefig(fig, figs_name)
% save(data_name, 'z_tds_m', 'z_tds_in', 'x_tds_m',
'x_tds_in',...
%     'Ra_tds_micron', 'Ra_tds_micro_in')

toc
```

VITA

Ryan Copenhaver was born in King, North Carolina to Richard and Pamela Copenhaver. Being the oldest child, he has three siblings: Haley, Royce, and Hayden. He received a BS in Mechanical Engineering from the University of North Carolina at Charlotte (UNC Charlotte) in December 2016. Ryan continued his academic studies in the doctoral program at UNC Charlotte. During his graduate studies at UNC Charlotte, he was a member of the High-Speed Machining Laboratory, Machine Tool Research Center, and the Center for Precision Metrology.

In May of 2019, Ryan followed the Machine Tool Research Center during its move to the University of Tennessee, Knoxville. While completing his research he continued to work in the Machine Tool Research Center at both the university's facilities and at Oak Ridge National Laboratory's Manufacturing Demonstration Facility. His research to date has focused on predicting the vibration behavior and surface finish during MTP turning using a custom lathe toolpath that ensures segmented chip generation.