AN ABSTRACT OF THE THESIS OF

Kaan Bahtiyar for the degree of Master of Science in Robotics presented on March 3, 2023.

Title: Detection of Chatter Vibrations and Adaptive Spindle Speed Optimization in Milling Process.

Abstract approved:

_____

Burak Sencer

This thesis focuses on improving productivity of machining processes via novel process monitoring and control techniques. Most machining processes suffer from vibrations in terms of forced and self-excited chatter vibrations. Although forced vibrations are inherent to the milling process itself, chatter vibrations are an unstable form of vibrations that destroy the work surface and affect machine tool's health. Accurate and timely detection of "chatter" and rapidly adjusting machining parameters to mitigate it is critical for productivity and longevity of the equipment. This work presents novel online chatter detection and adaptive spindle speed control strategies. Two chatter detection methods are presented to the literature. The first one is a real-time suitable lightweight approach, which monitors power spectrum of the vibration signal to detect chatter. The second method uses Principal Component Analysis (PCA) to provide a more accurate and robust detection. Both approaches are experimentally validated in milling. Next, an adaptive spindle speed regulation method is also developed which determines the optimal spindle speed to minimize total vibrations in milling. The approach bases on extremum seeking control (ESC), which can be implemented with or without a dither signal. Optimality of the ESC based adaptive spindle speed regulation approach is proven by analytically analyzing the milling process dynamics using semi-discrete time domain modeling. A time domain milling simulator is also developed to validate the functionality of the approach. It shows that

developed technique is effective in minimizing overall vibrations in milling to improve productivity.

Detection of Chatter Vibrations and Adaptive Spindle Speed Optimization in Milling Process

by
Kaan Bahtiyar

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented March 3, 2023
Commencement June 2023

Master of Science thesis of Kaan Bahtiyar presented on March 3, 2023

APPROVED:

Major Professor, representing Robotics

Associate Dean of Graduate Studies for the College of Engineering

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Kaan Bahtiyar, Author

ACKNOWLEDGEMENTS

I would like to express my gratitude to my advisor, Dr. Burak Sencer, for providing his supervision and support throughout my studies and research. It has been a pleasure working with such a hardworking and enthusiastic guidance.

I also like to thank my colleagues at Manufacturing Process Control Laboratory, my friends as well as the faculty and staff of School of Mechanical, Industrial and Manufacturing Engineering.

Finally, I would like to gratefully thank to my family for their endless support in my studies.

TABLE OF CONTENTS

Page

TABLE OF CONTENTS

Page

LIST OF FIGURES

Figure Page

LIST OF FIGURES

Figure Page

LIST OF FIGURES

Figure Page

LIST OF TABLES

Table Page

DEDICATION

I would like to dedicate this thesis to my dear parents Cavit and Meryem Bahtiyar, and my brother Selahittin Bahtiyar.

1 Introduction

Milling is one of the most commonly used manufacturing processes. Since they can produce consistent quality products with high material removal rate and tight geometric tolerances, milling machines are commonly used in many manufacturing industries. The process is very productive with great accuracy where smooth surface finishes are achieved.

Milling machines are arranged to generate relative motions between a cutter and a workpiece on multiple axes. As the most basic, it has three perpendicular axes, where the feed is on a plane perpendicular to cutter tool axis [1,2]. Additional rotational axes can be added to generate more complex parts. These machines can be horizontal or vertical, as well as manual or computer numerically controlled (CNC) [1]. Figure 1 shows a 3-axis industrial standard vertical milling machine 3D model. As shown in Figure 1, the cutter tool is mounted on a tool holder, and they both are attached to a rotating spindle which provides sufficient power for rotation of tool-tool holder and spindle assembly [1]. A workpiece attached to the milling table is fed toward to rotating cutter linearly [1,2].

In milling, the rotating cutter tool with one or more teeth removes material from the surface of the workpiece [2]. As the cutter rotates, each tooth on the tool removes a certain amount of material from the workpiece and generates chip during the motion of entering, cutting and leaving the workpiece [2]. This is repeated for each tooth on the cutter and each spindle revolution. Because of the dynamics of this cutting process, milling machines suffer from vibrations. If these vibrations exceed a certain level, they can cause damage to the surface finish, tool, and even the machine equipment itself .

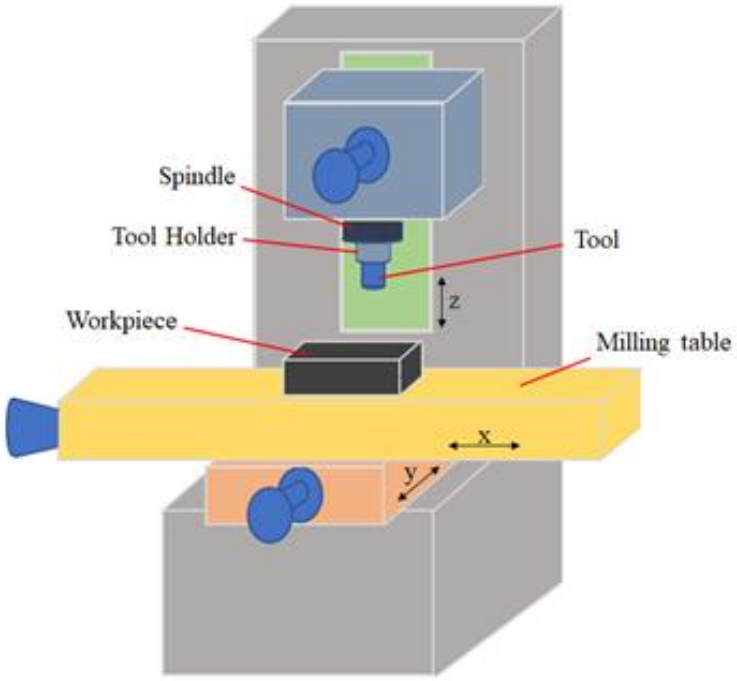

Figure 1: Example 3-axis vertical milling machine 3D model

In machining processes, there are 3 generalized vibration regimes which can be seen in Figure 2. First of all, if there is almost no vibrations in the machining process, the chip thickness does not fluctuate. In this case, the surface finish is very smooth and flat which is the ideal-case scenario. However, forced vibrations are inherent to the cutting process itself. They occur when there is a certain amount of vibrations in the process, and the chip thickness fluctuates steadily. The chip load is still constant in this scenario because the phase difference between previous and current tooth vibrations, i.e., the phase between upper and lower part of the chip is in-phase (see Figure 2). Therefore, the chip thickness fluctuates steadily. Forced vibrations are not critical for stability of the process, and the surface finish is still smooth and proper for most cases. However, when the process suffers from chatter vibrations, the chip thickness fluctuates violently. The phase between upper and lower part of the chip is out of phase which means chip thickness grows exponentially [2]. This is not stable, and once the process experiences chatter vibrations, these vibrations grow very quick and destroy the surface finish, tool,

and even machine equipment [3]. Therefore, prevention of chatter vibrations and providing stable cutting is very critical for milling processes. There are many studies have been done to avoid chatter vibrations. They can be separated into 2 groups.

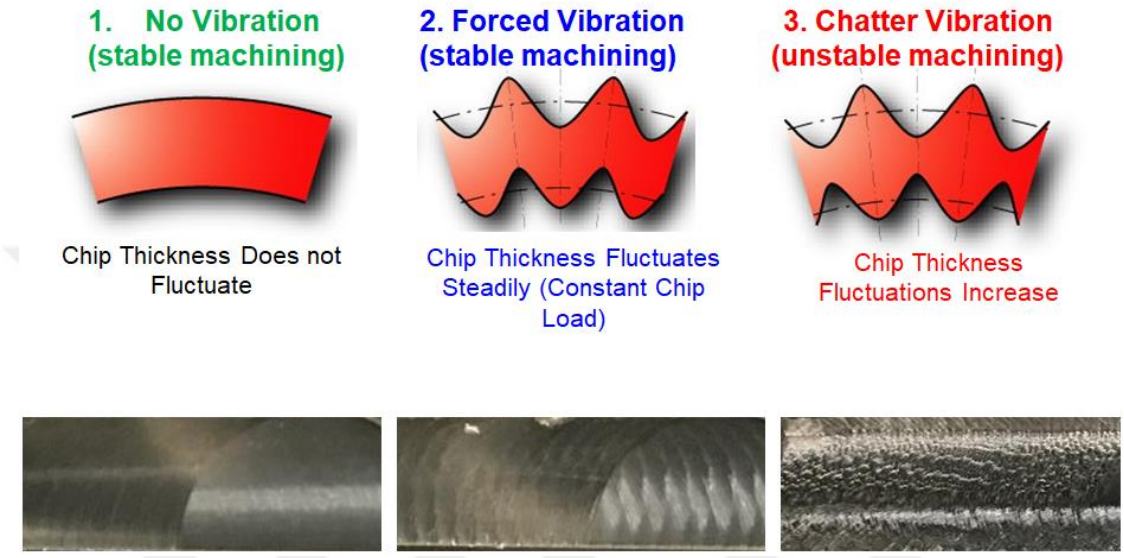

Figure 2: Chip thickness regimes in machining

First is based on chatter-free process planning. In this approach, the milling process is modeled before the cutting process is performed. It is proved that milling stability is dependent on spindle speed – depth of cut pairs [4] as will be detailed in chapter 3. In other words, there is a certain depth of cut for each spindle speed to cut material in stable manner. These limits are predicted by solving characteristic equation of the modeled milling process, and they are plotted on a diagram which shows critical depth of cut for each spindle speed, i.e., how much material can be removed for a specific spindle speed without experiencing chatter vibrations. These diagrams are called stability lobe diagrams (SLD). By selecting spindle speed – depth of cut pair underneath the diagram, stable machining condition is achieved. Figure 3 shows an example of SLD. The green region in the Figure 3 shows stable region while the red region represents unstable region. In other words, spindle speed – depth of cut pairs underneath the lobes, i.e., in green region, are stable machining conditions where it is unstable otherwise, in red region. As will be detailed in chapter 2, generating the SLD requires expert knowledge to model the process and analyze it, as well as pre-process

experiments to identify parameters used in the model. Therefore, many scholars and this thesis are focused on the following second method to obtain chatter-free machining.

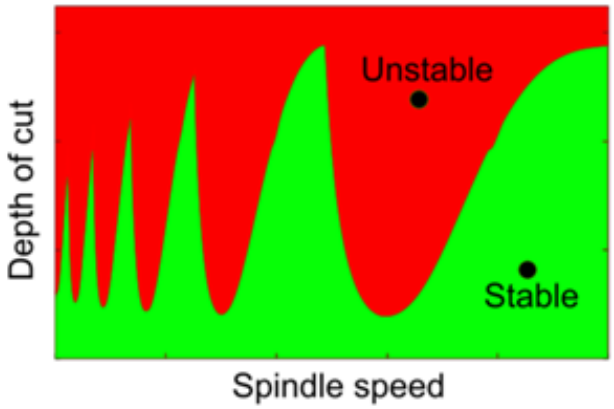

Figure 3: Stability lobe diagrams

The second approach is based on “online” detection of chatter vibrations during the machining process and mitigating (avoiding) them by adapting the cutting condition or stopping (terminating) the machining process before it harms the tool and the workpiece. This approach does not require any elaborate process modeling, which makes it favorable to use in practice and implementation on modern CNC machine tools. Vibration or sound signals during the milling process are acquired, and occurrence of chatter vibration is detected from time and/or frequency domain signal processing. The accuracy and the sensitivity of this online approach depends not only on the quality of the vibration signals, but also on how these signals are processed and evaluated. Low complexity milling process models needs to be introduced into the signal processing to more accurately. After detecting chatter vibrations in this approach, machining condition is changed to continue cutting with stable condition. The focus in this thesis lies on the spindle speed adjustment. Once the chatter detection algorithm provides an alarm, spindle speed can be adjusted stepwise in once, or it can be changed continuously until stable machining condition is achieved.

This thesis firstly proposes two novel online chatter detection algorithms: 1) Lightweight and 2) Robust online chatter detection algorithm. The lightweight online chatter detection method has low computational complexity making it suitable to run

on most low power real-time processors. The so-called Robust online chatter detection algorithm, on the other hand, it is designed for more robust in detecting occurrence of chatter. Thus, it provides more accuracy in detecting chatter for flexible workpieces/tools or noisy measurements. The results are benchmarked in time domain milling simulations and actual milling tests. Besides, this thesis also proposes two novel adaptive spindle speed optimization algorithms which mitigate overall vibrations during the process: 1) Conventional extremum seeking control (ESC)-based and 2) Dither-free extremum seeking control-based spindle speed optimization algorithm. Conventional ESC-based method is adjusting spindle speed at each time-sample. Due to hardware limitations, spindle speed controller may not be able to follow generated speed commands at every sampling time. This may generate wrong estimation of cost function map gradient. This problem is overcome with the second method, dither-free ESC-based spindle speed optimization algorithm. Calculation of cost function is adjusted so that more smooth speed commands are generated with stepwise changes at every pre-determined number of revolutions length of which can be adjusted based on spindle speed controller dynamics. Both methods are benchmarked in time-domain milling simulations.

The remainder of this thesis is divided into 5 chapters. Firstly, in chapter 2, a literature review on chatter vibrations, chatter-free process planning and online detection of chatter vibrations, spindle speed optimization and adjustment methods are presented. Chapter 3 contains systematic framework for dynamics of the milling process, milling chatter stability analysis and generation of stability lobes with different approaches, a time domain milling simulation which enables spindle speed change during any time step of the simulation and lastly vibration prediction based on pre-process modeling. Chapter 4 presents proposed chatter detection algorithms with their performance on time domain simulations and actual milling experiments. Chapter 5 presents adaptive spindle speed optimization algorithms and their performance on time domain milling simulations. The thesis is completed with the conclusion in chapter 6.

2 Literature Review

2.1 Introduction

Milling is one of the most important machining processes which is an intermittent cutting process using a cutter with one or more teeth [2]. A rotating spindle holds the cutter while the workpiece is moved toward the cutter linearly on the milling table [2]. It is very critical and productive for generating final part surfaces. While achieving high material remove rate, smooth surfaces that are functional and accurate are generated. Therefore, it is used in many industries such as aerospace, energy, automotive industries. This section of the thesis presents studies that have been done on chatter stability and how to mitigate vibrations in milling process.

2.2 Chatter Stability

Chatter stability in machining processes is long standing issue and attract lots of scholars. In particular, milling chatter stability is achieved either selecting stable machining parameters before the process or detecting chatter vibrations online once they occur during the process.

Altintas and Budak showed analytical prediction of chatter stability limits in milling based on a frequency domain approach which is one of the fundamental approaches in milling stability [4]. It is presented that for each spindle speed, there is a limit depth of cut which specifies maximum material that can be removed for that corresponding spindle speed. They generated stability limits which are used as main ground for pre-process model-based chatter avoidance by solving eigenvalue problem. For this purpose, they used frequency domain, assumed to consider average terms for directional coefficients, and named this zero-th order approach. By generating stability limits, named stability lobe diagrams, a stable pair of spindle speed – depth of cut can be selected so that process only exhibits cutting process inherent vibrations, i.e., forced vibrations. Later, further developments are performed on prediction of the stability limits by enhancing solution scope, such as multi-frequency solution [5] [6].

On the other hand, Stepan and Insperger proposed another approach to generate stability lobes by discretizing a spindle revolution and solving eigenvalue problem [7]. This method is so-called semi-discrete method which also attracts many scholars to analyze it [8–11]. This method is more accurate than the first method that Altintas and Budak proposed, but it is similar to later proposed multi-frequency solution of Altintas [6]. However, computational complexity of multi-frequency solution and semi-discrete method solution is much higher than firstly proposed method. This is because there is an iterative search for each spindle speed in these methods which yields more accurate solution while it takes much more time [5].

Besides of these methods, there are many studies to predict chatter stability limits. [12] proposed to use Bayesian machine learning to identify optimal stable cutting parameters. The learning algorithm learns a stability map based on test results. [13] used finite element analysis (FEA) to predict chatter vibrations. [14] used simulated annealing to predict stability limits. [15] proposed updated analytical prediction of stability lobes by using experimentally trained network model.

Although pre-process modeling and selecting stable conditions give accurate stability for the process, generating stability lobes requires several experiments and expert knowledge about machine dynamics and modeling. First of all, tool-workpiece specific constants (cutting force coefficients) are predicted by performing several stable cutting tests with different feedrates [16]. A least-squares fit is applied to these test results, and based on parameters of the fitting, coefficients for cutting model are predicted [2]. Besides, structural dynamics (tool-workpiece) which is represented as tool tip frequency response function (FRF) of the structure is needed to generate stability limits. Therefore, dynamic characteristic of the structure is obtained by hammer tests and modal analysis [1,2]. It should be noted that all these experiments consume time, need to be re-performed for different tools, workpieces, fixtures etc. Details of these processes will be given in the following chapter.

2.3 Online Chatter Detection

Instead of performing those pre-process modeling and experiments, many scholars focused on online detection of chatter vibrations which does not require any expertise, and pre-process experiments. After cutting starts with a spindle speed and depth of cut, spindle speed is adjusted if the chatter is detected in real-time.

Much research with physics-based approaches are proposed. [17] presents to seek chatter frequencies between spindle rotation frequency harmonics in the milling process from the overall spectrum. [18] proposed to track stable vibration via Kalman filters and use nonlinear energy operator to identify energy ratio. However, there are several tuning parameters in this method which is not easy to implement. [19] proposed similar approach with machine learning help. Their method is hybrid model where machine learning is trained to enhance the physics-based solutions especially for tool entry-exit regions. Also, several artificial intelligence-based approaches are proposed for chatter detection. [20] used experimental data of stable and unstable cutting and trained a self-organized network to detect chatter. However, artificial intelligence-based methods require chatter experiments dataset for training. Creating such database may be harmful for the machine equipment. Also, it is difficult to develop such a network unless they are used with physics of dynamic milling. [21] proposed to use autocorrelation coefficients between current and previous revolution's acceleration as indicator. If the chatter frequency is close to spindle frequency harmonics, their indicator misses chatter and generates index values close to stable one. [22] proposed to detect chatter via parametric modeling of the process. Although their recursive identification of the model performs well to identify increase in indexes they use, their detection threshold (variance of estimation error in their approach) may vary for different applications and needs to be re-tuned since it is not normalized for a simpler index. [23] developed a normalized chatter index by using nonlinear energy operator and Wigner distribution. [24] proposed a disturbance observer to detect chatter. Schmitz [25] used variance of sound signals, [26] proposed to use Poincare sections to distinguish chatter. Most method uses the ratio between stable vibrations and unstable vibrations to detect chatter where unstable vibrations are the components between

spindle rotation frequency harmonics in frequency spectrum. For this purpose, Fast Fourier transform of measured vibration or force signal is the most straightforward method [18]. However, Fourier transform based approaches have less resolution with small windows. On the other hand, larger windows add more delay to detection. Therefore, chatter component could be missed.

2.4 Spindle Speed Optimization

Once chatter is detected, spindle speed is changed to have stable machining. Here, most approaches are based on stopping the feed, and adjusting the speed. For this purpose, one of the most fundamental techniques is proposed by [17,27] where chatter frequency with the biggest amplitude is identified on frequency spectrum, and spindle speed is adjusted so that chatter frequency is coincided with a harmonic of updated spindle speed. [22] proposed similar technique. However, there is no guarantee that new spindle speed is a stable one. In the case that adjusted speed is also chattering, detection and adjusting processes are repeated until no chatter occurs [22]. This increases production time due to interrupting the feed.

Another approach to overcome chatter or mitigate vibrations is to disturb regenerative effect by changing spindle speed constantly [28]. However, spindle speed variation is not proper when the spindle speed is too high since the actuator of spindle requires to produce fast perturbation which may not be feasible because of actuation power.

[22] proposed another technique which lowers the total vibrations while stabilizing the process. However, their method requires estimation of chatter frequency and calculation of cost function produced by their detection algorithm. [29] proposed multi-rate chatter analysis and adaptive chatter avoidance. By performing chatter analysis, they produced an optimization problem based on semi-discrete method output parameters, and optimized spindle speed. However, their method has several tuning parameters as well as it requires modeling of the process.

2.5 Conclusion

Stable machining conditions can be achieved by selecting stable machining parameters through pre-process model-based approach or adjusting machining parameters after detecting chatter vibrations in real-time. Selecting parameters from stability charts requires dynamic model of the process, identification of modal parameters and constant coefficients through experiments which is time consuming and requires expertise. On the other hand, there are a lot of methodologies in the literature to improve detection and control of chatter vibrations in milling process. However, many of them requires lots of tuning parameters, modeling the process, or they are not robust enough against generating false alarms which limits productivity. For adjusting spindle speed after detecting chatter, some requires accurate prediction of chatter frequency. Moreover, there is no guarantee to adjust stable spindle speed. In this case, detection and adjustment is re-performed.

3 Milling Process Model

3.1 Introduction

This section presents modeling of the milling process and its dynamics. The stability of the process is examined through two different approaches, and selection of stable machining parameters through stability charts are explained.

3.2 Dynamics of Milling

Figure 4 shows a tool engaged to a workpiece in the milling process which is described as 2 DOF model. The equation of motion in milling can be written as follows:

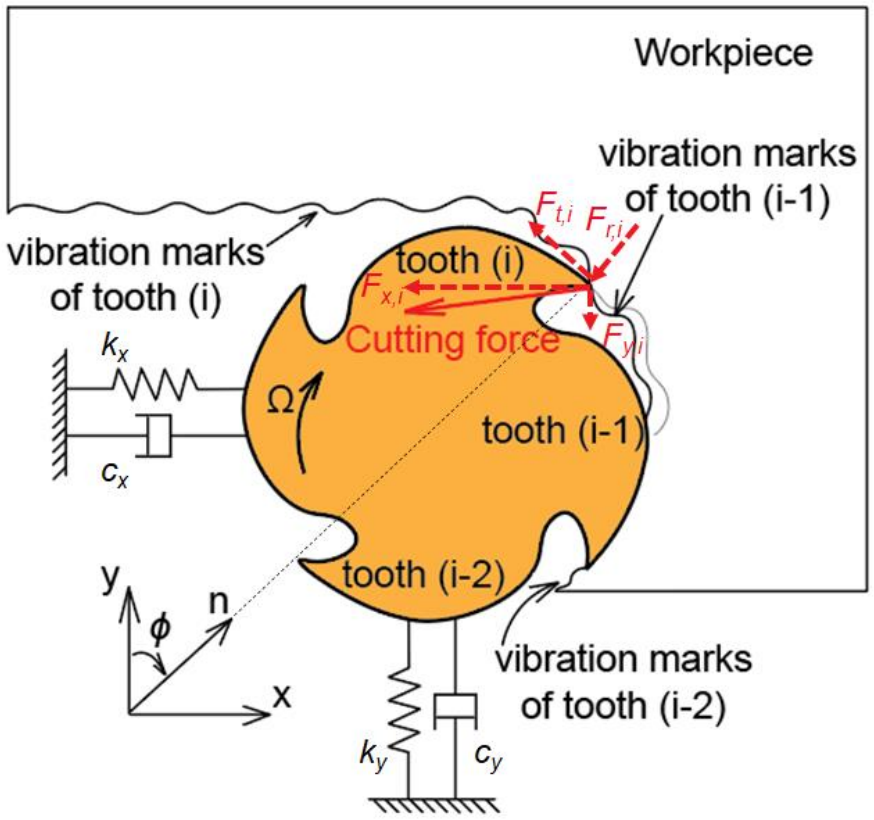

Figure 4: Tool-workpiece configuration

$$\boldsymbol{M\ddot{q}} + \boldsymbol{C\dot{q}} + \boldsymbol{Kq}(t) = \boldsymbol{F}(t) \qquad 1$$

where $\boldsymbol{F}$ is the cutting force vector, $\boldsymbol{q}$ is the dynamic displacement vector, and $\boldsymbol{M}$, $\boldsymbol{C}$, $\boldsymbol{K}$ are equivalent mass, damping and stiffness matrices respectively, in feed (x) and normal (y) directions which are expressed in Equation 2.

$$\left.\begin{aligned} \boldsymbol{F}(t) &= \begin{bmatrix} F_x(t) \\ F_y(t) \end{bmatrix} \\ \boldsymbol{q}(t) &= \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} \\ \boldsymbol{M} &= \begin{bmatrix} m_x & 0 \\ 0 & m_y \end{bmatrix} \\ \boldsymbol{C} &= \begin{bmatrix} c_x & 0 \\ 0 & c_y \end{bmatrix} \\ \boldsymbol{K} &= \begin{bmatrix} k_x & 0 \\ 0 & k_y \end{bmatrix} \end{aligned}\right\} \quad 2$$

Since cutting forces in milling are calculated by multiplication of the chip thickness and specific cutting force coefficients [2], instantaneous chip thickness is obtained to calculate cutting forces as follows:

$$h_i(t) = f_i sin\phi_i(t) + v_i(t-\tau) - v_i(t) \quad 3$$

where $h_i(t)$ is the instantaneous chip thickness, $f_i sin\phi_i(t)$ is the nominal chip thickness, f_i is feed per tooth, and ϕ_i is instantaneous angular immersion of tooth i, $v_i(t-\tau)$ and $v_i(t)$ are the previous and current tooth vibration contributions respectively. Since the chip thickness is measured in the radial direction (direction n in Figure 4), the contribution of current tooth to vibration is defined in radial direction as below [2]:

$$v_i(t) = -x(t)sin\phi_i(t) - y(t)cos\phi_i(t) \quad 4$$

where $x(t)$ is the dynamic displacement in feed direction, $y(t)$ is the dynamic displacement in normal direction, and instantaneous angular immersion varies with time as $\phi_i(t) = \Omega t$ where Ω is angular speed [rad/sec] [2]. The tooth period can be calculated as:

$$\tau = \frac{T}{N_t} \tag{5}$$

where T is the spindle period [sec], and N_t is the number of teeth. By writing equations of tangential and radial components of the resultant cutting force (see Figure 4) as follows:

$$\left.\begin{matrix} F_{t,i}(t) = g_i(t)(K_t b h_i(t)) \\ F_{r,i}(t) = K_r F_{t,i}(t) \end{matrix}\right\} \tag{6}$$

where K_t and K_r are constant cutting force coefficients, b is the axial depth of cut, $g_i(t)$ is a rectangular window function which is 1 if the i^{th} tooth is in cutting, and 0 otherwise, resultant dynamic force matrix is obtained by transforming those into global machine coordinates in feed and normal directions.

$$\left.\begin{matrix} F_x(t) = \sum_{i=0}^{N_t-1} -F_{t,i}cos\phi_i - F_{r,i}sin\phi_i \\ F_y(t) = \sum_{i=0}^{N_t-1} F_{t,i}sin\phi_i - F_{r,i}cos\phi_i \\ \boldsymbol{F}(t) = \begin{bmatrix} F_x(t) \\ F_y(t) \end{bmatrix} = \frac{1}{2} b K_t \boldsymbol{A}(t)[-\boldsymbol{q}(t) + \boldsymbol{q}(t-\tau) + \boldsymbol{h_0}(t)] \end{matrix}\right\} \tag{7}$$

where

$$\boldsymbol{h_0}(t) = \begin{bmatrix} c(t) \\ 0 \end{bmatrix} \tag{8}$$

and

$$\left.\begin{aligned} \boldsymbol{A}(t) &= \begin{bmatrix} a_{xx}(t) & a_{xy}(t) \\ a_{yx}(t) & a_{yy}(t) \end{bmatrix} \text{ where} \\ a_{xx}(t) &= -\sum_{i=0}^{N_t-1} g_i(t)\left[sin\left(2\phi_i(t)\right) + K_r\left(1 - cos\left(2\phi_i(t)\right)\right)\right] \\ a_{xy}(t) &= -\sum_{i=0}^{N_t-1} g_i(t)\left[1 + cos\left(2\phi_i(t)\right) + K_r\, sin\left(2\phi_i(t)\right)\right] \\ a_{yx}(t) &= \sum_{i=0}^{N_t-1} g_i(t)\left[1 - cos\left(2\phi_i(t)\right) - K_r\, sin\left(2\phi_i(t)\right)\right] \\ a_{yy}(t) &= \sum_{i=0}^{N_t-1} g_i(t)\left[sin\left(2\phi_i(t)\right) - K_r\left(1 + cos\left(2\phi_i(t)\right)\right)\right] \end{aligned}\right\} \quad 9$$

with $c(t)$ represents nominal chip thickness, $\boldsymbol{A}$ is the directional milling force coefficients matrix. The time lag between the current and the previous tooth $\tau(t)$, can be calculated from the Equation 10 [29] as:

$$\int_{t-\tau(t)}^{t} \omega_{sp}(t)dt = \frac{2\pi}{N_t} \quad 10$$

3.3 Chatter Stability of Milling

The regenerative effect, in other words, the delayed term in the milling process block diagram given in Figure 5 [30] causes the overall system to become unstable after a certain depth of cut for a specific spindle speed. Therefore, stability of the overall milling process depends on the spindle speed and depth of cut pairs. When the system becomes unstable, a vibration component, called chatter vibrations, appears, and grows exponentially which result in strong vibrations and becomes destructive for the workpiece, tool, and even machine equipment [1,2,31]. Therefore, chatter stability in milling process is one of the upmost significant which attracts lots of scholars.

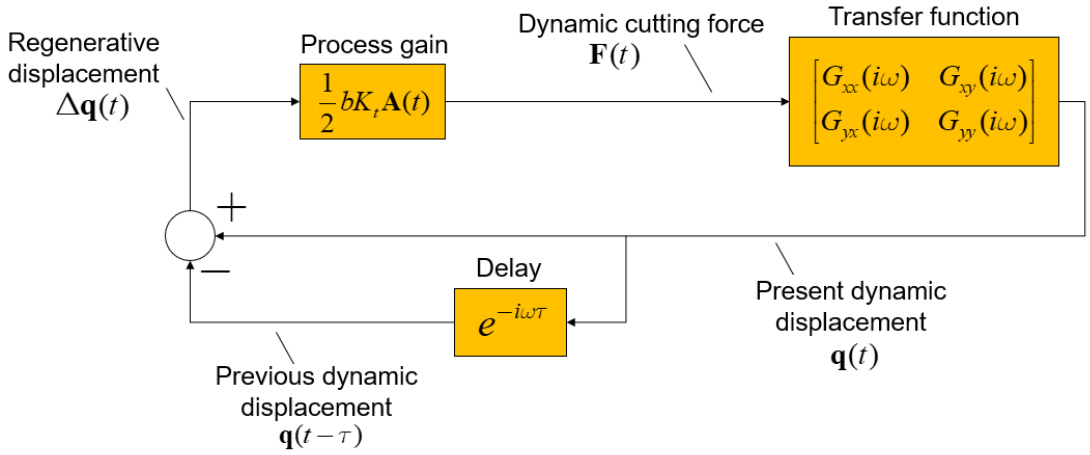

Figure 5: Block diagram of milling

As can be seen in Equation 7, as the tool rotates, the direction of the resultant force varies with time. However, besides of cutting forces, the directional cutting force coefficients are periodic at the tooth period. Therefore, stability of the whole process is analyzed by solving eigenvalues of the characteristic equation obtained based on dynamic milling expression given in Equation 7. Following sections present two different approaches to solve eigenvalue problem of the characteristic equation of the dynamic milling expressions.

3.3.1 Frequency Domain Solution: Zero-th Order Approach

Altintas [3,4] considered average component of the directional cutting force coefficients matrix Fourier series expansion to analyze milling stability problem in frequency domain. Firstly, the dynamic force expression given in Equation 7 is re-expressed in frequency domain as:

$$\{\boldsymbol{F}(\omega)\} = \frac{1}{2} bK_t \{\boldsymbol{A}(\omega)\} * \left(1 - e^{-i\omega\tau}\right)[\boldsymbol{G}(\omega)]\{\boldsymbol{F}(\omega)\} \qquad 11$$

where $*$ denotes convolution integral, $\boldsymbol{G}$ is the frequency response function (FRF) matrix of the machine (See Figure 5) [2,7,8]. The FRF matrix of the structure is defined at the cutter-workpiece contact zone [2]. In Figure 5, $G_{xx}(\omega)$ and $G_{yy}(\omega)$ are direct transfer functions in corresponding direction while $G_{xy}(\omega)$ and $G_{yx}(\omega)$ represents

cross-transfer functions. Stability of the process can be examined by checking if the dynamic given in Equation 11 is stable at frequency ω. Here, $\boldsymbol{A}(\omega)$ matrix given in Equation 11 can be approximated by its Fourier series expansion.

$$\boldsymbol{A}(\omega) = \sum_{k=-\infty}^{\infty} \boldsymbol{A_k}\, \delta(\omega - k\omega_{tp}) \tag{12}$$

where ω_{tp} is the tooth passing frequency, δ is the Dirac delta function. For the simplistic approximation, Altintas and Budak proposed to consider only average component of the Fouirer seris expansion, i.e., when k = 0 [2]. Then, Fourier transformation of $\boldsymbol{A}(t)$ matrix can be written as:

$$\mathcal{F}[\boldsymbol{A}(t)] = \sum_{k=-\infty}^{\infty} \boldsymbol{A_k}\, e^{ik\omega_{tp}t} \tag{13}$$

which has average component as follows.

$$\boldsymbol{A_0} = \frac{1}{\tau}\int_0^{\tau} \boldsymbol{A}(t)dt \tag{14}$$

This approach is valid only between the entry and exit angles of the tooth, in other words, if the tooth is in cutting ($g_i(t) = 1$). It becomes average value of directional cutting force coefficients matrix at the cutter pitch angle $\phi_p = 2\pi/N_t$, i.e., spindle rotation. The resultant average directional cutting force coefficients matrix becomes time invariant, but immersion dependent as follows [5]:

$$\left.\begin{aligned} \boldsymbol{A_0} &= \frac{N_t}{2\pi}\begin{bmatrix} a_{xx} & a_{xy} \\ a_{yx} & a_{yy} \end{bmatrix} \\ a_{xx} &= \frac{1}{2}[cos\,2\phi - 2K_r\phi + K_r\,sin\,2\phi]|_{\phi_{st}}^{\phi_{ex}} \\ a_{xy} &= \frac{1}{2}[-sin\,2\phi - 2\phi + K_r\,cos\,2\phi]|_{\phi_{st}}^{\phi_{ex}} \\ a_{yx} &= \frac{1}{2}[-sin\,2\phi + 2\phi + K_r\,cos\,2\phi]|_{\phi_{st}}^{\phi_{ex}} \\ a_{yy} &= \frac{1}{2}[-cos\,2\phi - 2K_r\phi - K_r sin2\phi]|_{\phi_{st}}^{\phi_{ex}} \end{aligned}\right\} \quad 15$$

Likewise, Fourier transform of the cutting force expression at the critical stability border, frequency ω_c, can be written and substituted into Equation 11 as follows [5]:

$$\begin{aligned} &\boldsymbol{F}(\omega) = \boldsymbol{P}_0\delta(\omega - \omega_c) \\ &\rightarrow \boldsymbol{F}(\omega) = \mathcal{F}(\{\boldsymbol{P}_0\}e^{i\omega_c t}) \end{aligned} \quad 16$$

where

$$\{\boldsymbol{P}_0\} = \frac{1}{2}bK_t\boldsymbol{A_0}\left(1 - e^{-i\omega_c\tau}\right)[\boldsymbol{G}(\omega_c)]\{\boldsymbol{P}_0\} \quad 17$$

and $\omega_c\tau$ is the phase shift between the previous and current vibrations. Therefore, by solving eigenvalues of this characteristic equation, stability of the milling process can be examined [2].

$$det(\boldsymbol{I} + \Lambda\boldsymbol{A_0}[\boldsymbol{G}(\omega_c)]) = 0 \quad 18$$

$$\Lambda = -\frac{N_t}{4\pi}bK_t\left(1 - e^{-i\omega_c\tau}\right) \quad 19$$

Eigenvalues yield to obtain critical axial depth of cut for a particular spindle period as follows [2]:

$$\left.\begin{aligned} b_{lim} &= -\frac{2\pi\Lambda_R}{N_t K_t}\left(1+\left(\frac{\Lambda_I}{\Lambda_R}\right)^2\right) \\ T &= \frac{1}{\omega_c}\left[(2k+1)\pi - 2\,tan^{-1}\left(\frac{\Lambda_I}{\Lambda_R}\right)\right] \end{aligned}\right\} \quad 20$$

where Λ_R and Λ_I are the real and imaginary component of the eigenvalue respectively.

3.3.2 Semi-Discrete Method

Averaged approximation for directional cutting force coefficients matrix ($\boldsymbol{A}$) is not sufficient to predict stability of the process in some cases such as small radial immersions and high spindle speeds [5]. In such cases, multi-frequency solution of chatter stability [5,6] or semi-discrete time domain solution is more appropriate for analysis. Here, semi-discrete method is examined which will be used latter for analysis of vibrations in adaptive spindle speed optimization section. Semi-discrete method provides higher accuracy but yields more complexity and computational time. However, a unified vibration analysis on the milling process can be examined through semi-discrete method [5].

Milling force model given in Equation 7 can be plugged into Equation 1 to obtain complete dynamic milling model.

$$\boldsymbol{M\ddot{q}} + \boldsymbol{C\dot{q}} + \boldsymbol{Kq}(t) = \frac{1}{2} bK_t \boldsymbol{A}(t)[-\boldsymbol{q}(t) + \boldsymbol{q}(t-\tau) + \boldsymbol{h_0}(t)] \quad 21$$

Linear periodically time-varying state space model of the milling with varying time delay is expressed as following [5,11,29]:

$$\dot{\boldsymbol{p}} = \boldsymbol{A}_1(t)\boldsymbol{p}(t) + \boldsymbol{A}_2(t)\boldsymbol{p}(t-\tau) + \boldsymbol{A}_3(t)\boldsymbol{u}(t) \quad 22$$

where

$$\left.\begin{gathered}\boldsymbol{p}(t)=\begin{bmatrix}\boldsymbol{q}(t)\\ \dot{\boldsymbol{q}}(t)\end{bmatrix}\\ \boldsymbol{u}(t)=\boldsymbol{h_0}(t)\\ \boldsymbol{A}_1(t)=\boldsymbol{A}_1(t+\tau)=\begin{bmatrix}\boldsymbol{0}_{2\times2} & \boldsymbol{I}_2\\ \boldsymbol{M}^{-1}(\delta\boldsymbol{A}-\boldsymbol{K}) & -\boldsymbol{M}^{-1}\boldsymbol{C}\end{bmatrix}\\ \boldsymbol{A}_2(t)=\boldsymbol{A}_2(t+\tau)=\begin{bmatrix}\boldsymbol{0}_{2\times2} & \boldsymbol{0}_{2\times2}\\ -\delta\boldsymbol{M}^{-1}\boldsymbol{A} & \boldsymbol{0}_{2\times2}\end{bmatrix}\\ \boldsymbol{A}_3(t)=\boldsymbol{A}_3(t+\tau)=\begin{bmatrix}\boldsymbol{0}_{2\times2}\\ -\delta\boldsymbol{M}^{-1}\boldsymbol{A}\end{bmatrix}\\ \boldsymbol{M}^{-1}=\begin{bmatrix}\frac{\omega_{nx}^2}{k_x} & 0\\ 0 & \frac{\omega_{ny}^2}{k_y}\end{bmatrix},\boldsymbol{C}=\begin{bmatrix}\frac{2\zeta_x k_x}{\omega_{nx}} & 0\\ 0 & \frac{2\zeta_y k_y}{\omega_{ny}}\end{bmatrix},\boldsymbol{K}=\begin{bmatrix}k_x & 0\\ 0 & k_y\end{bmatrix},\delta=\frac{1}{2}bK_t\end{gathered}\right\} \quad 23$$

Here ω_n, k, and ζ are natural frequency, stiffness, and damping ratio for direction $\{x,y\}$. If the spindle speed is assumed to be constant, time delay period can be obtained from Equation 10 as:

$$\tau=\frac{2\pi}{\omega_{sp}N_t} \quad 24$$

By dividing delay period τ into m number of discrete time intervals, $\tau=m\Delta t$ is obtained where Δt is the sampling period of the semi-discretization. Therefore, by averaging time-dependent and delay terms, state space representation is first approximated by considering initial condition of $\boldsymbol{p}(t_i)=\boldsymbol{p}_i$ as follows [11,29]:

$$\dot{\boldsymbol{p}}=\boldsymbol{A}_{1i}\boldsymbol{p}_i+\boldsymbol{A}_{2i}\boldsymbol{p}_{\tau i}+\boldsymbol{A}_{3i}\boldsymbol{u}(t) \quad 25$$

where

$$\left.\begin{gathered}\boldsymbol{A}_{ji}=\frac{1}{\Delta t}\int_{i\Delta t}^{(i+1)\Delta t}\boldsymbol{A}_j(t)dt \qquad (j=1,2,3)\\ \boldsymbol{p}_{\tau i}=\frac{1}{2}\boldsymbol{p}(t_i-\tau+\Delta t)+\frac{1}{2}\boldsymbol{p}(t_i-\tau)=\frac{1}{2}(\boldsymbol{p}_{i-m+1}+\boldsymbol{p}_{i-m})\end{gathered}\right\} \quad 26$$

for t_i is the current time, and $\boldsymbol{p}_i$ is the value of $\boldsymbol{p}(t_i)$. The discretization of the Equation 25 is then performed as follows:

$$\boldsymbol{p}_{i+1} = \boldsymbol{P}_i \boldsymbol{p}_i + \boldsymbol{R}_i \boldsymbol{p}_{\tau i} + \boldsymbol{Q}_i \boldsymbol{u}_i \tag{27}$$

where

$$\left.\begin{aligned} \boldsymbol{P}_i &= exp\,(\boldsymbol{A}_{1i}\Delta t) \\ \boldsymbol{R}_i &= (exp(\boldsymbol{A}_{1i}\Delta t) - \boldsymbol{I})^{-1}\boldsymbol{A}_{1i}^{-1}\boldsymbol{A}_{2i} \\ \boldsymbol{Q}_i &= (exp(\boldsymbol{A}_{1i}\Delta t) - \boldsymbol{I})^{-1}\boldsymbol{A}_{1i}^{-1}\boldsymbol{A}_{3i} \end{aligned}\right\} \tag{28}$$

Finally, it can be written in single-rate discrete state space equation as follows [29]:

$$\boldsymbol{z}[i+1] = \boldsymbol{D}_i \boldsymbol{z}[i] + \boldsymbol{E}_i \boldsymbol{u}[i] \tag{29}$$

where

$$\left.\begin{aligned} \boldsymbol{z}[i] &= [\boldsymbol{q}[i]^T \quad \dot{\boldsymbol{q}}[i]^T \quad \boldsymbol{q}[i-1]^T \quad \boldsymbol{q}[i-2]^T \quad \cdots \quad \boldsymbol{q}[i-m]^T]^T \\ \boldsymbol{D}_i &= \begin{bmatrix} \boldsymbol{P}_i & \boldsymbol{L}_i \\ \boldsymbol{B} & \boldsymbol{0}_{2\times 2m} \\ \boldsymbol{0}_{(2m-2)\times 4} & \boldsymbol{W} \end{bmatrix} \\ \boldsymbol{E}_i &= [\boldsymbol{Q}_i^T \quad \boldsymbol{0}_{2\times 2}^T \quad \cdots \quad \boldsymbol{0}_{2\times 2}^T]^T \\ \boldsymbol{L}_i &= \left[\boldsymbol{0}_{4\times(2m-4)} \quad \frac{1}{2}\boldsymbol{R}_i[1{:}\,4,1{:}\,2] \quad \frac{1}{2}\boldsymbol{R}_i[1{:}\,4,1{:}\,2]\right] \\ \boldsymbol{B} &= [\boldsymbol{I}_2 \quad \boldsymbol{0}_{2\times 2}] \\ \boldsymbol{W} &= [\boldsymbol{I}_{2m-2} \quad \boldsymbol{0}_{(2m-2)\times 2}] \end{aligned}\right\} \tag{30}$$

Here, $\boldsymbol{R}_i[1{:}\,4,1{:}\,2]$ are the elements in the range of 1st to 4th rows and 1st to 2nd of matrix $\boldsymbol{R}_i$. The (2m+4)-dimensional representation of discrete m-step state-space equation can be rewritten as follows:

$$\boldsymbol{z}[m(i+1)] = \boldsymbol{T}\boldsymbol{z}[mi] + \boldsymbol{S}\boldsymbol{U}[mi], \tag{31}$$

where

$$\left.\begin{array}{c} \boldsymbol{U}[mi] = [\boldsymbol{u}[mi]^T \quad \ldots \quad \boldsymbol{u}[mi+m-1]^T]^T \\ \boldsymbol{T} = \boldsymbol{\Phi}(m) \\ \boldsymbol{\Phi}(h) = \begin{cases} \boldsymbol{I}\ (h=0) \\ \boldsymbol{D}_{m-1} \ldots \boldsymbol{D}_{m-h}\ (otherwise) \end{cases} \\ \boldsymbol{S} = [\boldsymbol{\Phi}(m-1)\boldsymbol{E}_0 \quad \ldots \quad \boldsymbol{\Phi}(0)\boldsymbol{E}_{m-1}] \end{array}\right\} \quad 32$$

According to Floquet theory, stability of the overall process can be examined by looking at the maximum eigenvalues of the state transition matrix (***T***)**.** The process is asymptotically stable if the maximum eigenvalue for a corresponding spindle speed and depth of cut pair is less than 1 [5,11]. Here, by checking for whole spindle speed-depth of cut map with small increments, stability lobe diagrams can be obtained.

3.3.3 Generation of Stability Lobe Diagrams

The frequency domain solution (zero-th order approach) uses average periodic directional factors which reduces the stability solution into direct calculation of spindle speed and depth of cut [5]. Since there is no need for an iterative search, this reduces calculation time into a few seconds. However, in semi-discrete method, time varying directional factors are considered at each sampling time which predicts the stability limit at any spindle speed [5]. The accuracy of this method is dependent on the sampling interval [5]. Since the method predict stability iteratively, the computation time depends on the number of modes and sampling interval [5]. Therefore, semi-discrete method is more time consuming than zero order approach, but more accurate. It should be noted that for these methods, modal parameters such as modal mass, damping and stiffness should be identified as well as cutting force coefficients since they are used in both methods.

Firstly, tool-workpiece specific constants, i.e., cutting force coefficients, need to be predicted by performing several stable cutting tests with different feedrates [16]. For this prediction, a set of stable experiments at different feedrates are run [1,2,16]. For this, some low depth-of-cut tests are performed, and force data is measured by dynamometer. If the frequency spectrum of the force data is only concentrated at spindle rotation frequency and its harmonics, the test is accepted as stable [16]. Later,

several cutting tests are performed at constant immersion and depth of cut, but different feedrates, and force signal is measured. A least-squares fit is applied to average force components at each direction and feedrates. Based on parameters of the fitting, cutting and edge coefficients are predicted as given below [1,2,16].

$$\left.\begin{aligned} \overline{F_x} &= -K_{te}S + K_{re}T - \frac{f}{4}(K_{tc}P + K_{rc}Q) \\ \overline{F_y} &= -K_{te}T + K_{re}S + \frac{f}{4}(K_{tc}Q + K_{rc}P) \\ \overline{F_z} &= -\frac{bN_t}{2\pi}K_{ae}(\phi_{ex} - \phi_{st}) + fK_{Ac}T \end{aligned}\right\} \quad 33$$

where

$$\left.\begin{aligned} P &= \frac{bN_t}{2\pi} cos\, 2\phi|_{\phi_{st}}^{\phi_{ex}} \\ Q &= \frac{bN_t}{2\pi}\left[2\phi - sin\, 2\phi|_{\phi_{st}}^{\phi_{ex}}\right] \\ S &= \frac{bN_t}{2\pi} sin\, \phi|_{\phi_{st}}^{\phi_{ex}} \\ T &= \frac{bN_t}{2\pi} cos\, \phi|_{\phi_{st}}^{\phi_{ex}} \end{aligned}\right\} \quad 34$$

Here, ϕ_{ex} and ϕ_{st} are the start and exit angles of the cut which are π and 0 for slotting experiments, b is the axial depth of cut, f is the feedrate, $\overline{F_i}$ is the average force on direction i. For slotting experiments, these equations are expressed for a linear function of feedrates for each direction [1,2,16].

$$\overline{F_i} = \overline{F_{ie}} + f\overline{F_{ic}} \;\; (i = x, y, z) \quad 35$$

It should be noted that the predicted constants are valid for specific tool geometry and workpiece material [1,2]. Therefore, it should be re-performed if the cutter or the workpiece material is changed.

Secondly, identification of modal parameters is required to solve eigenvalue problems presented in previous sections and to generate stability lobes [2,32]. Identification of FRF is performed via modal testing techniques [2]. The vibratory response of the structure is measured to an impact force input [2]. Excitation with force is performed

through shakers or impact hammers. Since the amplitude and frequency of exerted force can be controlled via shaker controller, more accurate and high coherence excitation can be performed [1,2]. However, setting up a shaker on the machine is more time consuming [4]. Therefore, impact hammers are used in practice, which is easier to use, set up and quicker to perform measurement. For the experiment, a single axis accelerometer is attached on the structure, and an impact force with a hammer is applied on the same axis. This should be performed to the workpiece fixture as well especially if the workpiece is too flexible. The input force is measured via impact hammer, and output acceleration is measured via accelerometer. Modal analysis is performed as given in reference [2], and the resultant equation is given below.

$$G(j\omega) = \frac{X(ik\omega_r)}{F(ik\omega_r)} = \frac{X(i\omega)}{F(i\omega)} \tag{36}$$

Here, $G(i\omega)$ is the FRF of the machine on the measured axis, $X(i\omega)$ is the Fourier transform of the output vibration, $F(i\omega)$ is the Fourier transform of the force input, and $i\omega$ indicates spectra of complex numbers [2]. This can be further progressed by assumption of considering noisy sensor data which can be seen in reference [2]. It is shown that the effect of noise is attenuated by considering cross-power spectrum. It should be noted that the FRF is in the form of force input to acceleration output which can be converted to displacement by dividing $(i\omega)^2$. Finally, dominant modes are selected, and a transfer function is fitted to experimental modes to generate modal parameters [2].

Both methods presented in previous sections calculate the stability limit for each spindle speed which shows how much depth of cut can be achieved at most for a specific spindle speed. The diagrams show those limits are so called stability lobe diagrams. Table 1 shows an example milling process parameters which are used to generate stability lobes given in Figure 6. It should be noted that, identification of parameters given in Table 1 is performed as explained above (i.e., cutting force coefficient identification experiments, hammer test and modal analysis). Figure 6 shows SLD that are generated by using both methods, zero-th order approach, and semi

discrete method. As can be seen, there is a small difference between two methods. As investigated by Merdol and Altintas [5], the difference between two methods becomes more obvious for highly intermitted cutting processes, and zero-th order approach is not able to capture all the stability pockets (lobes).

Table 1: Milling process parameters

Modal parameters	$k_x = 40 \times 10^6 \left[\frac{N}{m}\right], k_y = 55 \times 10^6 \left[\frac{N}{m}\right]$ $\omega_{nx} = 1200\ [Hz], \omega_{ny} = 1000\ [Hz]$ $\zeta_x = 0.05, \zeta_y = 0.02$
Cutting parameters	$N_t = 4$ $K_t = 707.2016 \left[\frac{N}{mm^2}\right]$ $K_r = 168.8654 \left[\frac{N}{mm^2}\right]$

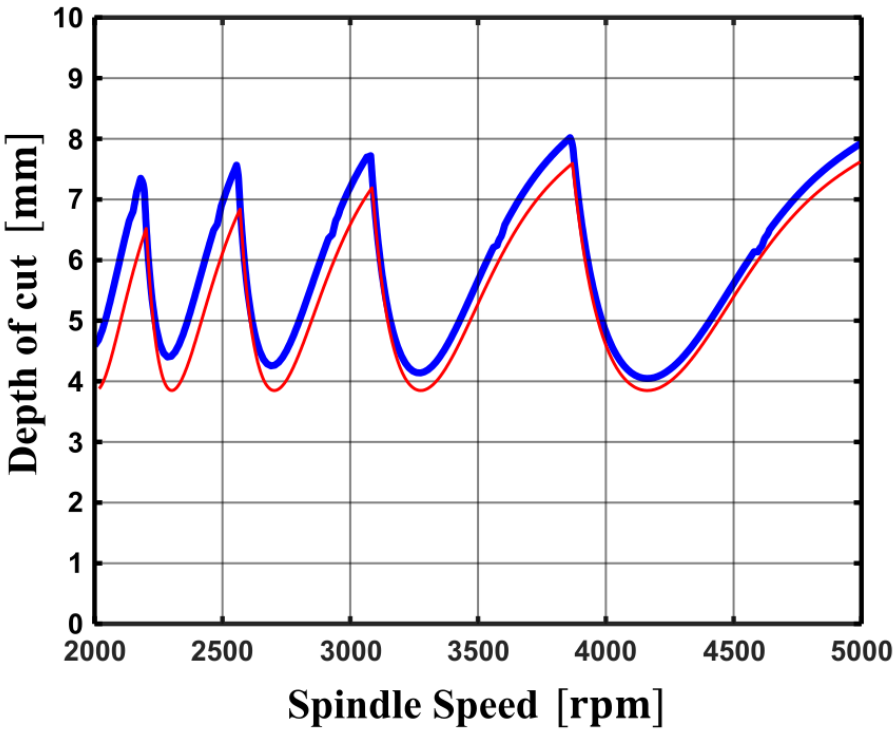

Figure 6: Stability lobes generated by zero-th order approach (red) and semi-discrete method (blue)

3.4 Time Domain Milling Simulation

Time domain simulation enables to analyze time-dependent variation of dynamic systems since it enables to solve the coupled, time-delay equations of motion numerically. It is an effective approach for analysis of milling dynamics since it includes nonlinearity occurs such as tool jump out which is not accounted in semi-discrete method. Besides, it enables to analyze all dynamics with respect to time such as local cutting forces, tool vibration for the corresponding cutting conditions including transient responses as opposed to analytical stability lobes or semi-discrete method [1,2]. In order to have accurate results from time domain simulation, the regenerative effect of surface waviness caused by the vibration of the previous tooth passing should be considered. The procedure to obtain displacement vector is as follows.

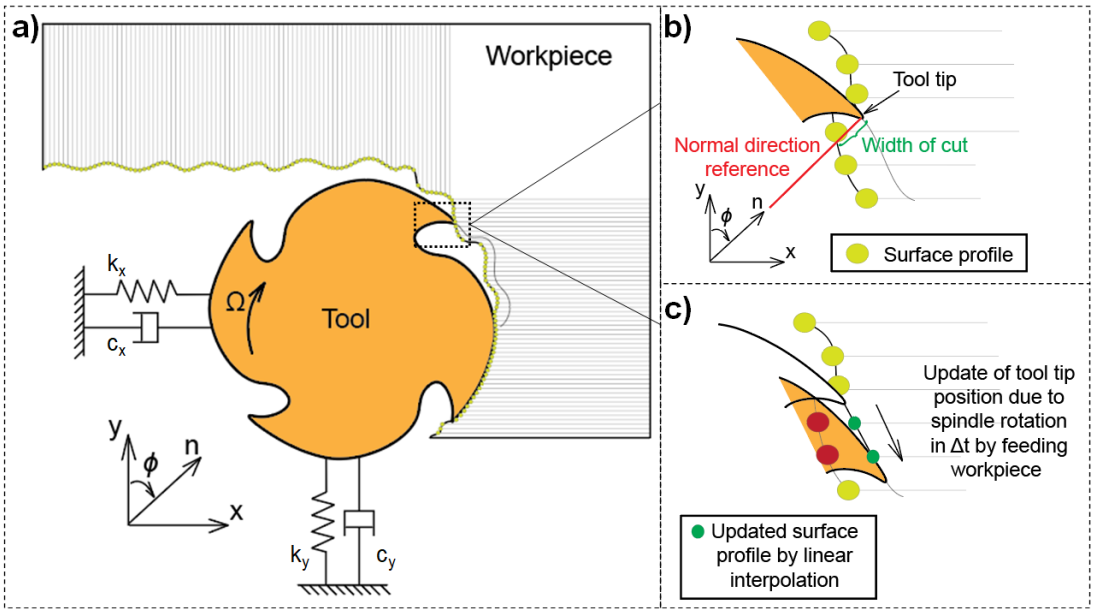

Figure 7: Time domain simulation model steps

First of all, simulation is created with a surface profile method as given in Figure 7a. The instantaneous chip thickness is obtained as shown in Figure 7b and Equation 3. Tangential and radial direction cutting forces are calculated based on instantaneous chip thickness as given in Equation 6. These forces are transformed into the global machine coordinates (*x*, *y*), and all teeth contributions are summed to express total

dynamic milling force acting on the cutter as given in Equation 7. New dynamic displacements in feed and normal directions are obtained by numerical solution of the following ordinary differential equations.

$$\left.\begin{matrix} m_x\ddot{x} + c_x\dot{x} + k_x x = F_x(t) \\ m_y\ddot{y} + c_y\dot{y} + k_y y = F_y(t) \end{matrix}\right\} \quad 37$$

where m is the modal mass, c is the modal viscous damping coefficient, and k is the modal stiffness. If the tool has multiple modes, above equations are obtained for each tool mode and the results are summed in each direction [1,2,33]. Solution for these ordinary differential equations can be obtained by any numerical method such as Forward Euler or Tustin Method.

If the tool has helix, it is sectioned into a number of axial slices where each slice is considered as straight teeth cutter [33]. Each slice incorporates as distance delay from the previous slice. The required slice width is given in [1,33] as follows:

$$db = \frac{D\ d\phi}{2tan\ (\gamma)} \quad 38$$

where D is the cutter diameter, db is the thickness of each slice, γ is the helix angle. The process is repeated by incrementing tool rotation angle. From simulation perspective, to calculate vibration contributions, simulation is performed by keeping surface profile. Once the chip thickness, cutting forces and dynamic displacements are obtained, cutting tool is rotated by $\Delta\phi$ based on a fixed sampling time Δt where the relationship is given as follows as can be seen in Figure 7b-c.

$$\Delta\phi = \frac{2\pi\Delta t}{T} \quad 39$$

This strategy is proposed to perform a spindle speed optimization in real-time on the simulation (to be used in chapter 5). Here, there is no guarantee that increment in the tool rotation results in the tool position to be coincide with a pre-specified surface

profile since the surface profile is obtained and fixed for initial spindle speed which varies with time after optimization starts. Therefore, surface profile is obtained by having a higher resolution than the initial spindle period as follows:

$$surface\ angles = linspace(0,2\pi, resolution * \frac{T}{\Delta t}) \quad 40$$

Higher resolution increases simulation time, and lower resolution may cause inaccurate results for higher spindle speeds. As the tool rotates, current tool positions are incremented by $\Delta\phi$ and the surface profile between current and previous position of the current tooth are updated by linear interpolation. This enables the user to change spindle speed during any time step of the simulation. Overall steps for the simulation can be seen in Table 2, and Figure 7.

Table 2: Time domain simulation steps

1- Simulation is initialized for a fixed time step Δt.
   a. Surface profile is created by the equation in Equation 40.
   b. Modal parameters of the tool, cutting force coefficients for the tool-workpiece pair is set.
   c. Cutting condition is set (depth of cut, immersion, spindle speed, etc.)
   d. Initial teeth locations on the created surface profile is set.
2- Instantaneous chip thickness is obtained by linear interpolation and cutting forces and dynamic displacements are calculated by solving differential equations given in Equation 37.
3- Tool edge position is updated by rotating the tool $\Delta\phi$ and the profiles between previous and current tooth edge positions are interpolated.
4- The process is repeated until simulation end time.

3.5 Forced Vibration Prediction Based on Semi-Discrete Method

[29] proposed an approach to identify forced vibrations based on the semi-discrete analysis. Since the state variable given in the Equation 31 includes the component in X

and Y directions, any directional vibration can be evaluated. Here, since the surface roughness is dependent on the Y (normal) directional amplitude [29], the output equation of the state space is written by expressing followings.

$$\boldsymbol{y}[mi] = [y[mi] \quad \dots \quad y[mi-m]]^T = \boldsymbol{O}\boldsymbol{z}[mi] \tag{41}$$

where

$$\boldsymbol{O} = [\boldsymbol{o}_2^T \quad \boldsymbol{o}_6^T \quad \boldsymbol{o}_8^T \quad \cdots \quad \boldsymbol{o}_{2m+4}^T]^T \tag{42}$$

where $\boldsymbol{o}_i \in \mathbb{R}^{1\times(2m+4)}$ i^{th} component of the vector $\boldsymbol{o}_i \in \mathbb{R}^{1\times(2m+4)}$ is 1 and the others are 0. Here, the state space systems transfer function matrix can be described as follows:

$$\boldsymbol{G_f}[z] = \boldsymbol{O}(z\boldsymbol{I} - \boldsymbol{T})^{-1}\boldsymbol{S} \tag{43}$$

Since the eigenvalues of $\boldsymbol{T}$ is related to chatter stability, [29] proposed singular values of $\boldsymbol{G_f}$ is related to forced vibration, i.e., DC component of the vibration since the input of this state space representation is the static chip thickness. Also, the expression can be defined as $\boldsymbol{G_f} = \boldsymbol{O}(\boldsymbol{I} - \boldsymbol{T})^{-1}\boldsymbol{S}$ if the static chip thickness is set as constant. Here, with the maximum singular values of the DC component, $\boldsymbol{G_f}$, forced vibration analysis can be performed as well. However, it should be noted that, since the semi-discrete model does not include nonlinearity such as tool jump out, accurate relative vibration level analysis on the chattering (unstable) conditions may not be performed.

In this section, tool vibrations in milling process are examined by using time domain milling simulator and semi-discrete method analysis. For this purpose, parameters of an example milling model are given in Table 1. By using those, time-domain milling simulation produces cutting forces and tool vibrations in x (feed) and y (normal) directions. On the other hand, by using semi-discrete method, stability lobes for corresponding milling model can be found by eigenvalues map of state transition

matrix and forced vibration related information can be obtained by singular values map of the DC component.

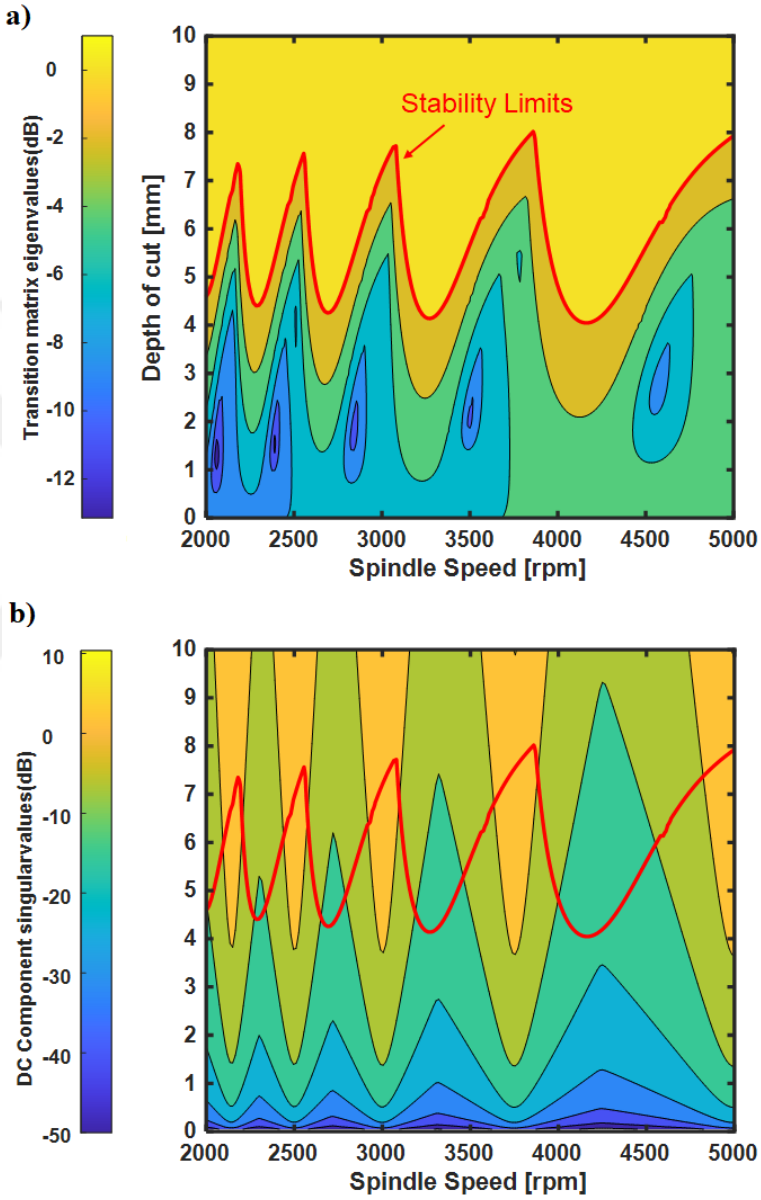

Figure 8: Results of semi-discrete method analysis with stability lobe diagrams (red curves), a) transition matrix eigenvalues, b) DC component singular values

Figure 8 presents stability lobes diagram (red curves) of the milling process given in Table 1 where the limit depth of cut (b_{lim}) is determined when $abs(\text{eig}(\boldsymbol{T})) = 1$. Stable machining conditions are obtained by selecting spindle speed-depth of cut pairs

underneath the stability limits in Figure 8. The semi-discrete method [29] can also be used to evaluate relative forced vibration level. Forced vibration level is controlled by the structural dynamics, cutting conditions and commanded chip thickness. Singular values of DC (static) component of $\boldsymbol{G_f = O(I - T)^{-1}S}$ represent relative forced response of the system. Figure 8 also presents eigenvalue and singular value maps of the given machining process.

For detailed analysis, consider the operating points P1-P3 in Figure 9a. The red curve shows stability lobes and black curve presents DC component singular values at 3 mm depth of cut in Figure 9a. All points, P1-P3, are stable since they are under the stability limits. However, they exhibit different forced vibration levels. Based on singular values of $\boldsymbol{G_f}$, P1 shows largest forced vibration whereas P3 has the smallest forced vibrations. Figure 9b shows simulation results for those conditions. Note that there is 2-3x difference in the forced vibration level depending on those operating conditions.

Next, consider the operating conditions in Figure 10a. The red curve shows stability limits while black and blue curves present DC component singular values and transition matrix eigenvalues at 5 mm depth of cut, respectively. Both P4 and P5 are in the stability lobe itself, and thus they are stable. However, depending on the spindle speed, they exhibit different forced vibration levels, which is again predicted by the singular values of $\boldsymbol{G_f}$. Time domain simulations are conducted by starting the spindle speed from an unstable point (S) and altering to the operating points P4 and P5 as shown Figure 10b. Note that depending on the magnitude of $abs(\text{eig}(\boldsymbol{T}))$, the system stabilizes faster or slower. For instance, P4 has smaller eigenvalue and thus chatter stabilizes faster. It should be noted from that eigenvalue and singular value pair indicate characteristics of each operating point, and they can be used to optimize the machining system response as presented in chapter 5.

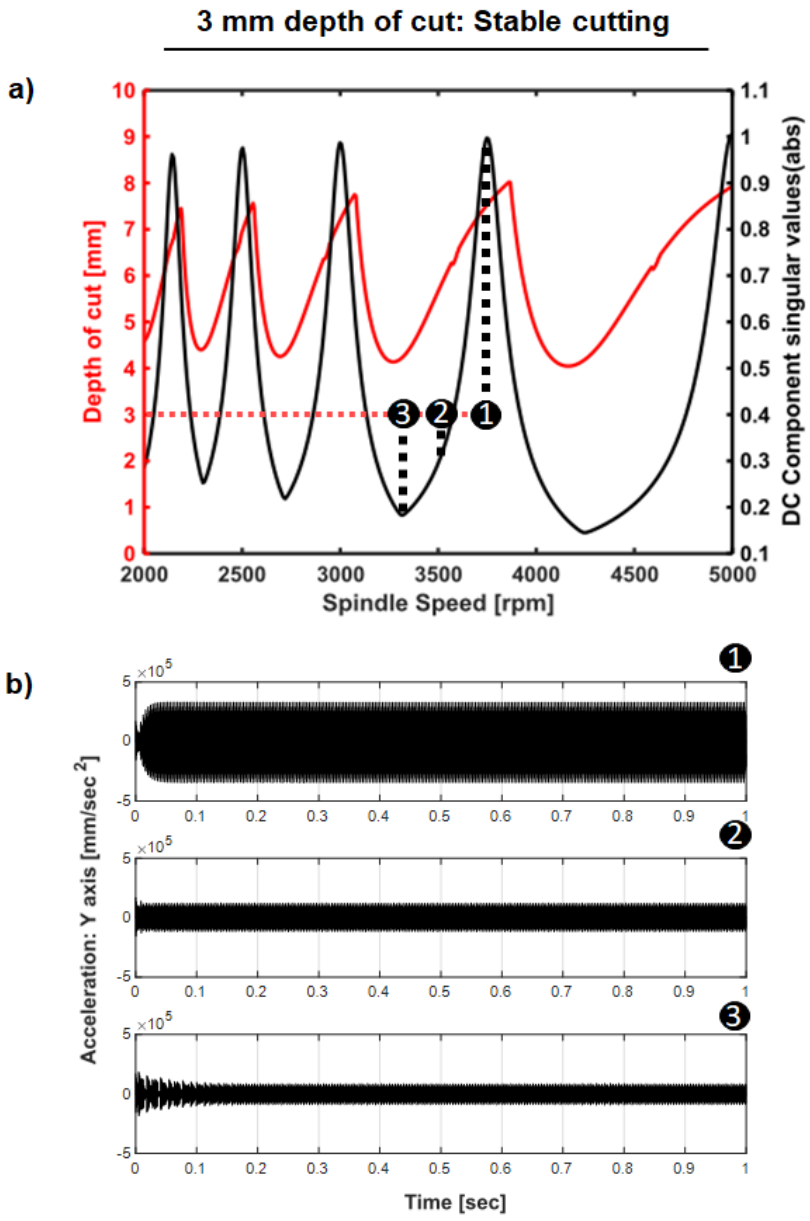

Figure 9: Analysis of semi-discrete method with time domain milling simulation, stable condition

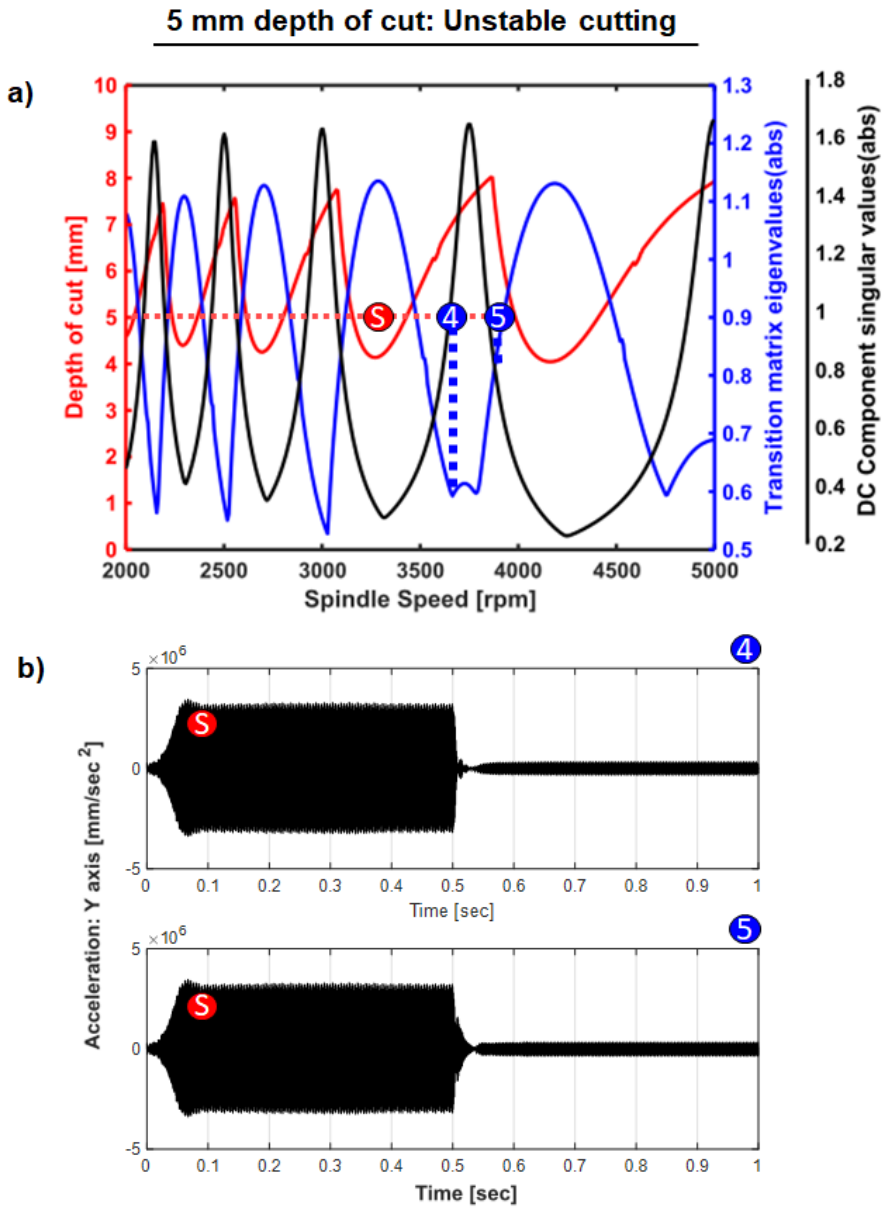

Figure 10: Analysis of semi-discrete method with time domain milling simulation, unstable condition

Lightweight and Robust Chatter Detection Algorithms for Milling

Kaan Bahtiyar and Burak Sencer

Manufacturing Letters

Volume 33, Supplement, 2022, Pages 388-394

4 Online Chatter Detection

4.1 Introduction

Most machining processes suffer from unwanted vibrations in the form of forced and self-excited chatter vibrations that deteriorate the manufacturing quality [1,2,31,34]. Once triggered, chatter vibrations may grow up to be destructive and damage both the workpiece, the tool, and the machine components [1,2]. Previous section presented dynamics of the milling process, and state-of-the-art to prevent chatter vibrations by selecting stable machining conditions based on SLD. However, modeling whole milling process, identifying cutting force coefficients, modeling machine dynamics, and predicting stability lobes are lengthy processes. Besides, these processes require expertise and expert knowledge which may not be applicable always. Therefore, online detection of chatter vibrations once they occur and adjusting the process parameters on-the-fly to conserve the work part is critical for machine shops [2]. This requires rapid and accurate detection of onset of chatter in milling, which has been a long-standing issue.

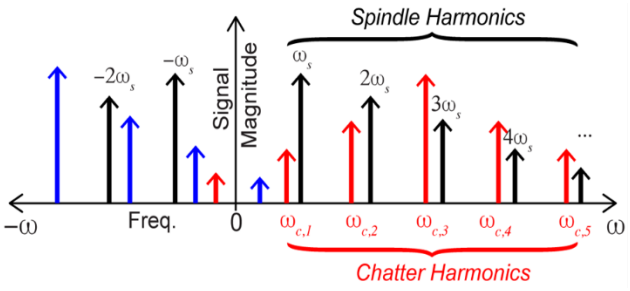

Figure 11: Spindle synchronized forced and chatter frequency harmonics

Typically, vibration signals in machining are measured via accelerometers attached either on the spindle or the workpiece, or machining sound signals acquired by a microphone [18,35]. All these approaches are effective in measuring the vibratory behavior and their frequency spectra in machining assuming the sensor has a linear transfer function [18,35]. Figure 11 illustrates frequency spectrum of vibration in milling process. As observed, frequency spectrum of tool/workpiece vibration exhibits various harmonics that are associated with the forced and chatter vibrations

components. Forced vibrations are inherent to the cutting process. Due to the imperfection at the tool-tool holder-spindle assembly, forced vibrations are typically observed periodic at spindle's angular rotation frequency [31].

Chatter vibrations are; however, self-excited and once triggered, they develop near structural resonances [2]. Depending on the structural flexibilities and process parameters, chatter vibrations may exhibit a single fundamental frequency and/or several side bands [18]. Thus, when chatter occurs, the total measured vibration signal (x) in milling can be represented as:

$$x(nT) = \underbrace{\sum_{i=1}^{P} A_{f,i}(nT)\, sin\left(i\omega_{sp}nT + \phi_{f,i}\right)}_{x_f(nT)} + \underbrace{\sum_{m=1}^{M} A_{c,m}(nT)\, sin\left(i\omega_{c,m}nT + \phi_{c,m}\right)}_{x_c(nT)} + v(nT) \quad (44)$$

where x_f is the forced vibration component truncated by its *P* dominant harmonics with the spindle rotation frequency ω_{sp}, x_c is the chatter component approximated by *M* harmonics with the chatter frequency ω_c, $A_{f,i}$, $A_{c,m}$ and $\phi_{f,i}$, $\phi_{c,m}$ are amplitude and phase of i^{th} periodic and m^{th} chatter components respectively, v is the measurement noise, *n* is the sample counter and *T* is the sampling time. Note that as mentioned, above signal can be measured by an accelerometer or a microphone [9]

Since forced vibrations are inherent to the process, one way to detect occurrence of chatter is to test if the vibration frequency spectrum is corrupted by any chatter harmonics/components. Current state of the art uses Fourier transform (FT) of vibration signal acquired by accelerometers or microphones [2,36] and searches for chatter harmonics in-between forced vibration harmonics [37]. If the detected chatter vibration magnitude is above a threshold value, process is deemed to be "chattering". However, timely detection of the chatter in real-time is an issue. Frequency domain approaches

are based on application of the Short-time Fourier transform (STFT), which processes vibration signals within a limited window. Large window (batch) sizes to evaluate the Fourier transform add delay to the detection. On the other hand, small window size limits resolution of frequency analysis, and thus chatter component could be missed all together.

To robustly detect chatter, another approach is to consider the spectral power/energy ratio of forced and chatter vibration components. The signal power/energy is defined by its frequency domain power spectrum [38]. For instance, if the chatter vibration signal power or its ratio over total vibration signal power is over a pre-defined threshold, the process is dominated by chatter, and chatter is detected [18,19,39,40]. This threshold is defined as the power ratio (*PR*). Such approach is more robust than only using chatter magnitude to detect chatter. However, it still provides false alarms in detecting chatter [41].

In this thesis, two power-ratio based chatter detection methods are provided. The first method is a computationally efficient real-time suitable frequency domain approach, which computes total and forced vibration energies, and employs power ratio to detect chatter. The second method is based on Principal Component Analysis (PCA), which provides greater accuracy in detecting harmonic components in the signal and computes the power ratio selectively which leads to more robust and timely detection of chatter. Performance of the methods are benchmarked in actual milling tests.

4.2 Lightweight Chatter Detection Algorithm

At first, a computationally efficient method is presented, and power ratio approach is applied to detect occurrence of chatter. The idea is to compute the total vibration signal power in milling and then determine the chatter power by subtracting the forced vibration signal power component where “power” is defined as the power spectrum of the vibration signal measured through accelerometers. Occurrence of chatter is detected when greater than 50% of the total vibration power is attributed to the chatter vibration

component. In other words, when chatter vibration power spectrum exceeds the forced vibration power spectrum, chatter starts to dominate the whole process and it is detected.

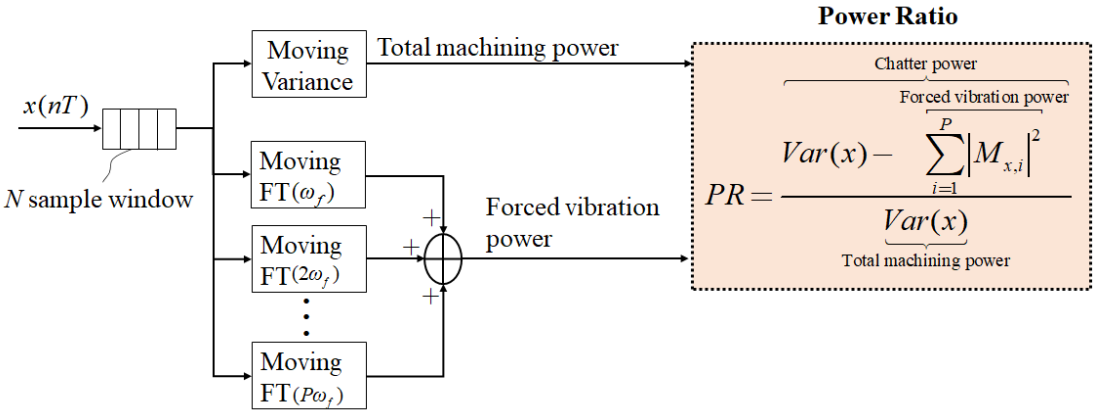

Figure 12: MV-MFT algorithm block diagram.

Figure 12 shows the block diagram of the approach. As shown, measured vibration signal is buffered with N sample window and moving variance operator is applied to compute total signal power while moving Fourier transforms of P dominant forced vibration harmonics with the spindle rotation frequency (ω_{sp}) in Equation 44 are applied to compute forced vibration power. By subtracting forced vibration power from total machining power, chatter vibration power is calculated and the ratio of chatter power over total machining power is used to detect occurrence of chatter. The following sections explain the operations in detail as follows.

4.2.1 Total Vibration Signal Power

Firstly, total power/energy of a signal is defined in frequency domain as:

$$E(X) = \frac{1}{2\pi}\int_{-\pi}^{\pi} |X(\omega)|^2 d\omega \qquad 45$$

where X(ω) is the discrete-time Fourier transform (DTFT) of the signal at ω frequency, and E is the energy of the signal. Notice that taking DTFT at high resolution along

entire frequency spectrum of a signal and evaluating Equation 45 is not efficient for real-time implementation. Instead, variance operator (Var) is used to calculate energy of the signal in time domain. Variance of a time domain signal presents its cumulative power spectrum or overall energy except the static component [38,42,43] as:

$$E(X) = Var(x) = \frac{1}{N}\sum_{n=0}^{N-1}(x[n] - E[x])^2 \qquad 46$$

where x is the signal, n is the sample counter and N is the total number of samples in the window, E is the expectation operator. Since vibration (acceleration) signals in milling are sinusoidal with zero mean, expectation operator does not affect the outcome. Signals are typically categorized as "energy" or "power" signals and since vibration signals in milling are sinusoidal, they have infinite energy and thus defined as power signals [44]. Therefore, rather than energy, signal power is considered for acceleration signals in milling, and their variance correspond directly to signal power. As shown in Figure 12, signal variance is computed in limited window and depending on the window size (N), computational cost may increase. In order to reduce the computational effort, moving variance (MV) computation is adapted from [39]. This can be interpreted as first-in, first-out logic. When the variance evaluation window slides one sample, effect of the removed sample is subtracted from the variance calculation for the window and effect of the new sample is added. As a result, Equation 46 does not need to be applied repetitively for each sample. Instead, it is re-written in following form [39]:

$$Var(x) = \underbrace{\frac{1}{N}\sum_{i=n}^{n+N-1}(x[i])^2}_{E[x^2]} \underbrace{-2\frac{1}{N}\sum_{i=n}^{n+N-1}(x[i]E[x]) + \frac{1}{N}\sum_{i=n}^{n+N-1}(E[x])^2}_{-(E[x])^2} \qquad 47$$

which can be simplified in recursive form as:

$$Var(x_{cur}) = Var(x_{prev}) + \frac{x(n+N+1)^2}{N} - \frac{x(n)^2}{N} - \left(\frac{x(n+N+1)}{N} - \frac{x(n)}{N}\right) \quad 48$$

As shown, Equation 48 reduces the number of arithmetic operations and makes total signal power computation more efficient for real-time implementation on an NC system.

4.2.2 Forced Vibration Component Power

Previous section presented time-domain approach to compute total vibration signal power, which contains stable forced and unstable chatter vibrations when chatter occurs. This section focuses on extracting forced vibration power based on its harmonics at the spindle rotation frequency. As shown in Figure 12, forced vibration power is computed by evaluating its FT only at limited spindle harmonics. Furthermore, similar to the moving variance approach, computational complexity is reduced by adapting a moving Fourier transform (MFT) strategy [39] as follows.

Firstly, *N*-sample discrete Fourier transform (DFT) of vibration signal (*x*) can be written as:

$$\left.\begin{aligned} X_k[n] &= \sum_{m=0}^{N-1} x[n+m] e^{\left(-\frac{j2\pi km}{N}\right)} \\ X_k[n+1] &= \sum_{m=0}^{N-1} x[n+m+1] e^{\left(-\frac{j2\pi km}{N}\right)} \end{aligned}\right\} \quad 49$$

where X represents complex DFT of the signal, n and m are sample counters, and j is the complex number. It should be noted that k is the frequency counter, and by setting $k = RPM \times i/60$ for $i = 1, \dots, P$, DFT of the signal at selected spindle harmonics can be extracted where P is the number of spindle rotation frequency harmonics in Equation 44, and RPM stands for the spindle speed in rotation per minute. Equation 49 applies

DFT in a window with *N* samples. Similar to moving variance method presented in previous section, calculation of DFT power spectrum can be put in a recursive form to reduce computational effort [39], and called as the moving Fourier transform. Based on Equation 49, the following can be written.

$$M_k[n] = \sum_{m=0}^{N-1} x[n+m] e^{\left(-\frac{j2\pi k(n+m)}{N}\right)} = R_k[n] + jI_k[n] \tag{50}$$

where *M* is the MFT, *R* is the real part and *jI* is the imaginary part. By comparing the definition of DFT, MFT value has a phase lag to the DFT value as:

$$M_k[n] = \sum_{m=0}^{N-1} x[n+m] e^{\left(-\frac{j2\pi km}{N}\right)} e^{\left(-\frac{j2\pi kn}{N}\right)} = X_k[n] e^{\left(-\frac{j2\pi kn}{N}\right)} \tag{51}$$

However, as can be seen, the power spectrum of MFT is equal to that of DFT.

$$|M_k[n]|^2 = \left| X_k[n] e^{\left(-\frac{j2\pi kn}{N}\right)} \right|^2 = |X_k[n]|^2 \underbrace{\left| e^{\left(-\frac{j2\pi kn}{N}\right)} \right|^2}_{=1} = |X_k[n]|^2 \tag{52}$$

MFT ignores the phase characteristic of the analyzed signal, but it calculates power spectrum with fewer computations. The update of the MFT is as follows:

$$\left.\begin{aligned} R_k[n] &= R_k[n-1] - x[n]\cos\left(\frac{2\pi kn}{N}\right) + x[n+N]\cos\left(\frac{2\pi k(n+N)}{N}\right) \\ I_k[n] &= I_k[n-1] - x[n]\sin\left(\frac{2\pi kn}{N}\right) - x[n+N]\sin\left(\frac{2\pi k(n+N)}{N}\right) \end{aligned}\right\} \tag{53}$$

Then, the MV and MFT algorithms are combined as shown in Figure 12 to identify chatter. Power ratio of the total measured vibration signal and the forced vibration component are compared as indicator of chatter onset. Notice that the power of total signal and power of the forced vibration part are calculated separately. Therefore, power ratio is defined as:

$$PR = \frac{\overbrace{Var(x) - \overbrace{\sum_k |M_k|^2}^{Forced\ vibration\ power}}^{Chatter\ power}}{\underbrace{Var(x)}_{Total\ machining\ power}} \quad (54)$$

where *x* denotes the total vibration signal, *k* is spindle frequency harmonics encountered to obtain forced vibration part. Equation 54 defines $PR \in [0,1]$. When $PR = 0$, the process is fully stable. In other words, signal power contains only forced vibrations. In contrary, when $PR = 1$, the signal is completely dominated by chatter. When chatter occurs, both forced and chatter vibrations become apparent. Therefore, power ratio of $PR = 0.5$ is selected as the threshold to detect chatter. It should be noted that, for $PR > 0.5$, over half of the vibration signal power is associated with chatter.

4.3 Robust Chatter Detection Algorithm

Principal component analysis (PCA) is used to reduce the dimensionality of a dataset by keeping most important original variables [45]. By identifying main factors accounting for the variances, it extracts general trends in the data in descending significance order [46]. This characteristic of the PCA can be used to extract the frequency components of the vibration signal and their dominancy or significance over the total signal variance.

4.3.1 Principal Component Analysis Based Chatter Detection

PCA [45,47] is used to detect forced and chatter power components in Equation 44 more accurately and robustly as follows.

Firstly, it is shown that spectral power of a sinusoidal signal can be computed based on its covariance matrix. Consider a single sinusoidal signal with amplitude *A*, frequency ω, and phase ϕ sampled at T.

$$x(nT) = Asin(\omega nT + \phi) \tag{55}$$

The Hankel matrix ($\boldsymbol{H}$) of such a sampled sinusoidal signal is constructed with p rows and q columns as:

$$\boldsymbol{H} = A\begin{bmatrix} sin(0\omega T + \phi) & \cdots & sin\big((q-1)\omega T + \phi\big) \\ \vdots & \ddots & \vdots \\ sin\big((p-1)\omega T + \phi\big) & \cdots & sin\big((p+q-2)\omega T + \phi\big) \end{bmatrix}_{p\times q} \tag{56}$$

and can be re-written in Euler form as:

$$\boldsymbol{H} = \frac{A}{2j}e^{\phi j}\begin{bmatrix} e^{0\omega Tj} & \cdots & e^{(q-1)\omega Tj} \\ \vdots & \ddots & \vdots \\ e^{(p-1)\omega Tj} & \cdots & e^{(p+q-2)\omega Tj} \end{bmatrix}_{p\times q} - \frac{A}{2j}e^{-\phi j}\begin{bmatrix} e^{-0\omega Tj} & \cdots & e^{-(q-1)\omega Tj} \\ \vdots & \ddots & \vdots \\ e^{-(p-1)\omega Tj} & \cdots & e^{-(p+q-2)\omega Tj} \end{bmatrix}_{p\times q} \tag{57}$$

where it can be observed that both matrices in Equation 57 are rank 1 and based on that it can be proved that $\boldsymbol{H}$ would have 2 non-zero eigenvalues [47]. The covariance matrix ($\boldsymbol{C}$) of $\boldsymbol{H}$ can then be constructed as:

$$\boldsymbol{C} = E[(\boldsymbol{H} - E[\boldsymbol{H}])(\boldsymbol{H} - E[\boldsymbol{H}])^T] = \begin{bmatrix} cov(\boldsymbol{x_1}, \boldsymbol{x_1}) & \cdots & cov(\boldsymbol{x_1}, \boldsymbol{x_m}) \\ \vdots & \ddots & \vdots \\ cov(\boldsymbol{x_m}, \boldsymbol{x_1}) & \cdots & cov(\boldsymbol{x_m}, \boldsymbol{x_m}) \end{bmatrix} \tag{58}$$

where $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{iq}]$. Eigenvalue decomposition on $\mathbf{C}$ is applied,

$$\boldsymbol{C}\boldsymbol{\alpha}_i = \lambda_i\boldsymbol{\alpha}_i \rightarrow \sum_{i=1}^{h}\boldsymbol{C}\boldsymbol{\alpha}_i\boldsymbol{\alpha}_i^T = \sum_{i=1}^{h}\lambda_i\boldsymbol{\alpha}_i\boldsymbol{\alpha}_i^T \tag{59}$$

where λ_i are eigenvalues, $\boldsymbol{\alpha}_i$ are eigenvectors of $\mathbf{C}$, and h is the number of eigenvalues. Knowing that $\boldsymbol{H}$ has 2 eigenvalues for a single tone (harmonic) sinusoidal signal in Equation 55, $\boldsymbol{C}$ must have 2 non-zero eigenvalues for each frequency component, which can be represented as:

$$\boldsymbol{CI_m} = \lambda_1 \boldsymbol{\alpha}_1 \boldsymbol{\alpha}_1^T + \lambda_2 \boldsymbol{\alpha}_2 \boldsymbol{\alpha}_2^T \tag{60}$$

By defining the energy of covariance matrix as sum of square of its elements as in [47,48], it can be shown that the square root of sum of square of its eigenvalues is proportional to square of the sinusoidal signal's amplitude:

$$\begin{aligned} E(\boldsymbol{C^2}) = \lambda_1^2 \underbrace{\sum_{i=1}^{m} \boldsymbol{\alpha}_{i,1}^2}_{=1} \underbrace{\sum_{i=1}^{m} \boldsymbol{\alpha}_{i,1}^2}_{=1} + \lambda_2^2 \underbrace{\sum_{i=1}^{m} \boldsymbol{\alpha}_{i,2}^2}_{=1} \underbrace{\sum_{i=1}^{m} \boldsymbol{\alpha}_{i,2}^2}_{=1} \\ + 2\lambda_1 \lambda_2 \underbrace{\sum_{i=1}^{m} \boldsymbol{\alpha}_{i,1} \boldsymbol{\alpha}_{i,2}}_{=0} \underbrace{\sum_{i=1}^{m} \boldsymbol{\alpha}_{i,1} \boldsymbol{\alpha}_{i,2}}_{=0} = \lambda_1^2 + \lambda_2^2 \end{aligned} \tag{61}$$

where $\mathrm{E}(\boldsymbol{C^2})$ is energy of the covariance matrix for signal in Equation 55. Equation 61 indicates that energy of the covariance matrix $\boldsymbol{C}$ can be found as sum of its eigenvalues' squares. Also, the covariance matrix is obtained proportional to sinusoidal signal's amplitude square (A^2) by substituting Equation 56 into Equation 58. Therefore, larger amplitude of a frequency component yields larger sum of squared eigenvalues in covariance matrix which means bigger energy of covariance matrix.

Table 3: PCA-based chatter detection

1- Choose *p*, *q* and *L*, form Hankel matrix $\boldsymbol{H}$ using measured periodic signal *x*.
2- Calculate covariance matrix $\boldsymbol{C}$ of Hankel matrix $\boldsymbol{H}$.
3- Perform eigen decomposition and obtain eigenvalues λ & eigenvectors $\boldsymbol{\alpha}$, and sort eigenvalues in descending order.
4- Select dominant eigenvalues λ_{sel} and their corresponding eigenvectors $\boldsymbol{\alpha}_{sel}$ by using Equation 62.
5- Perform DFT on selected eigenvectors and cluster selected forced vibration and chatter vibration related eigenvalues $\lambda_{f,sel}$, $\lambda_{c,sel}$
6- Calculate the power ratio by using Equation 63.

The analysis applied on a single sinusoidal (harmonic) signal can be extended to a multi-harmonic signal to compute overall signal power [48]. The algorithm given in Table 3 explains the steps of proposed algorithm. Firstly, p x q size Hankel matrix of measured vibration signal is generated, and the corresponding covariance matrix is constructed as shown in Equation 58. Eigenvalue decomposition is applied to extract all the eigenvalues and eigenvectors of the covariance matrix. The proposed algorithm only considers number of dominant harmonics for the signal power calculation by setting a cut-off ratio (L) defined as:

$$L = \frac{\sum_{i=1}^{l} \lambda_i}{\sum_{j=1}^{p} \lambda_j} \tag{62}$$

where $L \in [0,1]$ is the so called "contribution rate", l is the number of dominant eigenvalues, p is the total number of eigenvalues. In general, the cumulative contribution rate is set a certain value (0.8-0.95) to de-noising or data compression applications [47]. This is because the number of eigenvalues is determined by the Hankel matrix size which generates (p-$2w$) noise related eigenvalues, where w is number of variables within the data (sinusoidal harmonics for chatter application). When contribution rate is set to $L = 1$, all the eigenvalues are considered in the power calculation. For $L = 0$, no eigenvalue is considered. $L = 0.5$ is used to ensure that only 50% of the sum of eigenvalues are considered. Since some frequency components have more power than others in cutting process, the contribution rate helps to identify the dominant frequencies, which leads filtering of weak components and noise. This enables to reject both weak forced vibrations and chatters. The effect of this strategy is discussed in the experimental results section.

Then, Fourier transform of eigenvectors corresponding to selected dominant eigenvalues are used to obtain their frequencies. The frequencies are compared with spindle rotation frequency harmonics, and all the eigenvalues/vectors are classified for forced or chatter vibration related. Power ratio is then calculated by sum of squared chatter related eigenvalues over sum of all selected eigenvalues squares as:

$$PR = \frac{\sum_{i=1}^{r} \lambda_{c,sel,i}^2}{\sum_{j=1}^{l} \lambda_{sel,j}^2} \quad (63)$$

where l is the number of selected eigenvalues, r is the number of chatter related ones of selected eigenvalues. Overall block diagram of the approach is given in Figure 13.

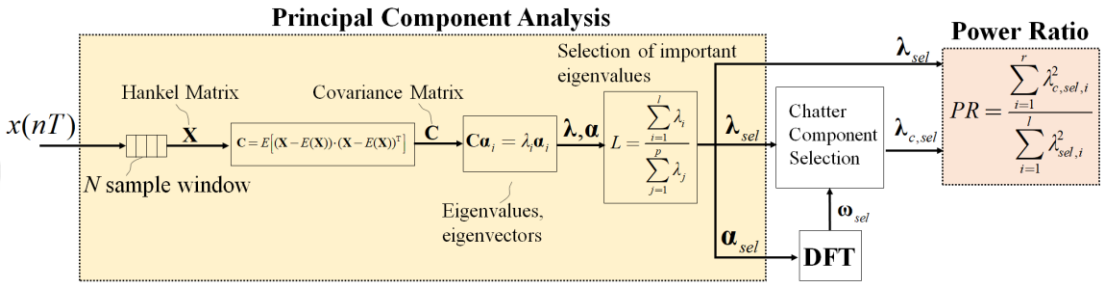

Figure 13: Principal component analysis-based chatter detection algorithm block diagram

4.4 Simulation Results

Proposed algorithms are firstly performed on time domain milling simulation. Milling model parameters given in Table 1, and 4 flute 19 mm diameter cutter is used in the simulation, and acceleration of the tool is used to detect occurrence of the chatter. Figure 14 shows SLD of the process. Full immersion (slotting) simulations are conducted at 5 mm depth of cut with stepwise change of spindle speed i) 3750 rpm to 4200 rpm, ii) 3750 rpm to 3250 rpm. Sampling frequency of the simulation is 10 kHz, and window size (N) is set to 7 spindle periods for both algorithms. Hankel matrix size of the PCA method is set to 5 spindle period x 2 spindle period. Power ratio threshold is set to 0.5 for detection of chatter in both algorithms, and the contribution rate (L) for the PCA method is set to 0.6, i.e., 60% of the sum of eigenvalues are considered for selection step.

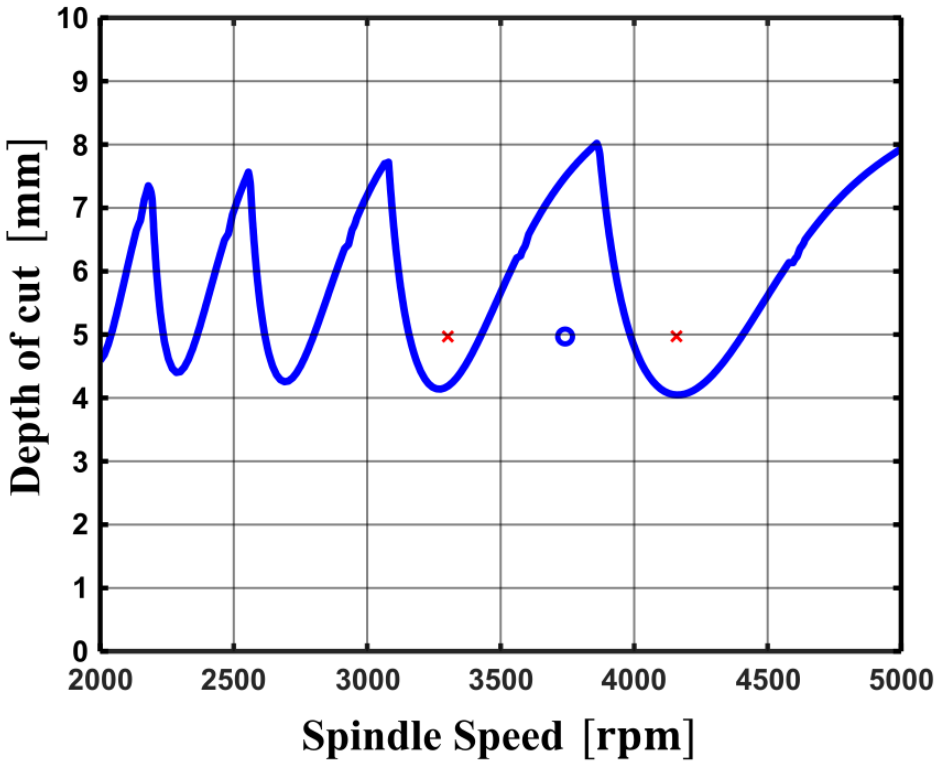

Figure 14: Stability lobe diagram of simulated milling process

Figure 15 and Figure 16 shows tool acceleration and power ratio processed by proposed algorithms. In Figure 15, simulation is started at a stable spindle speed – depth of cut pair of 3750 rpm and 5 mm depth, and spindle speed is changed to an unstable one, 4200 rpm, with a sudden step increase at 0.5th second. In Figure 16, it is shown when the spindle speed is changed to 3250 rpm from 3750 rpm. As shown in both figures, both algorithms timely detect chatter. Once the process becomes unstable, PCA-based algorithm provides 100% power ratio while MFT algorithm provides closer values to 100%. This is because thanks to contribution rate, PCA-based algorithm considers component of the vibration signal only if they are dominant enough to be included power calculation. This enables to filter out weak forced vibration and chatter components and the noise. On the other hand, MFT algorithm considers whole power spectrum. Therefore, it does not provide 100% power ratio.

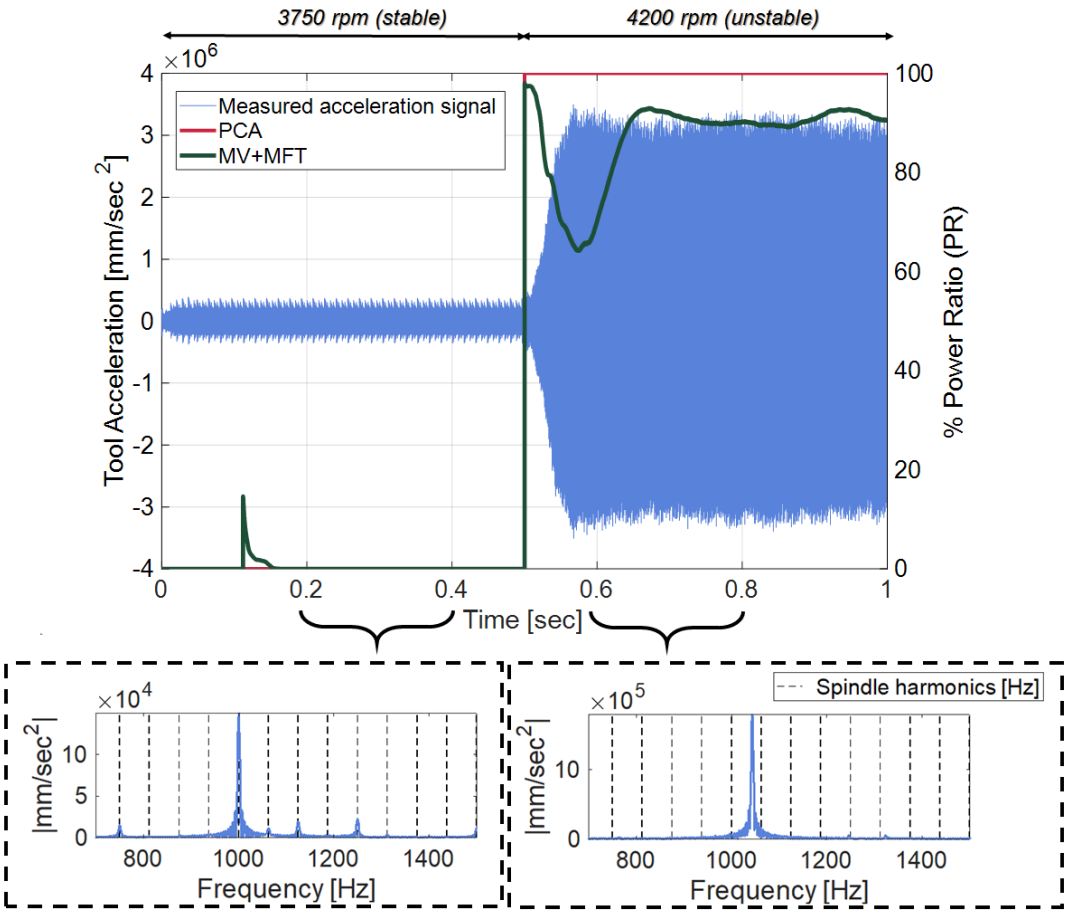

Figure 15: Simulation validation of proposed algorithms, 5 mm depth of cut, 3750 rpm to 4200 rpm

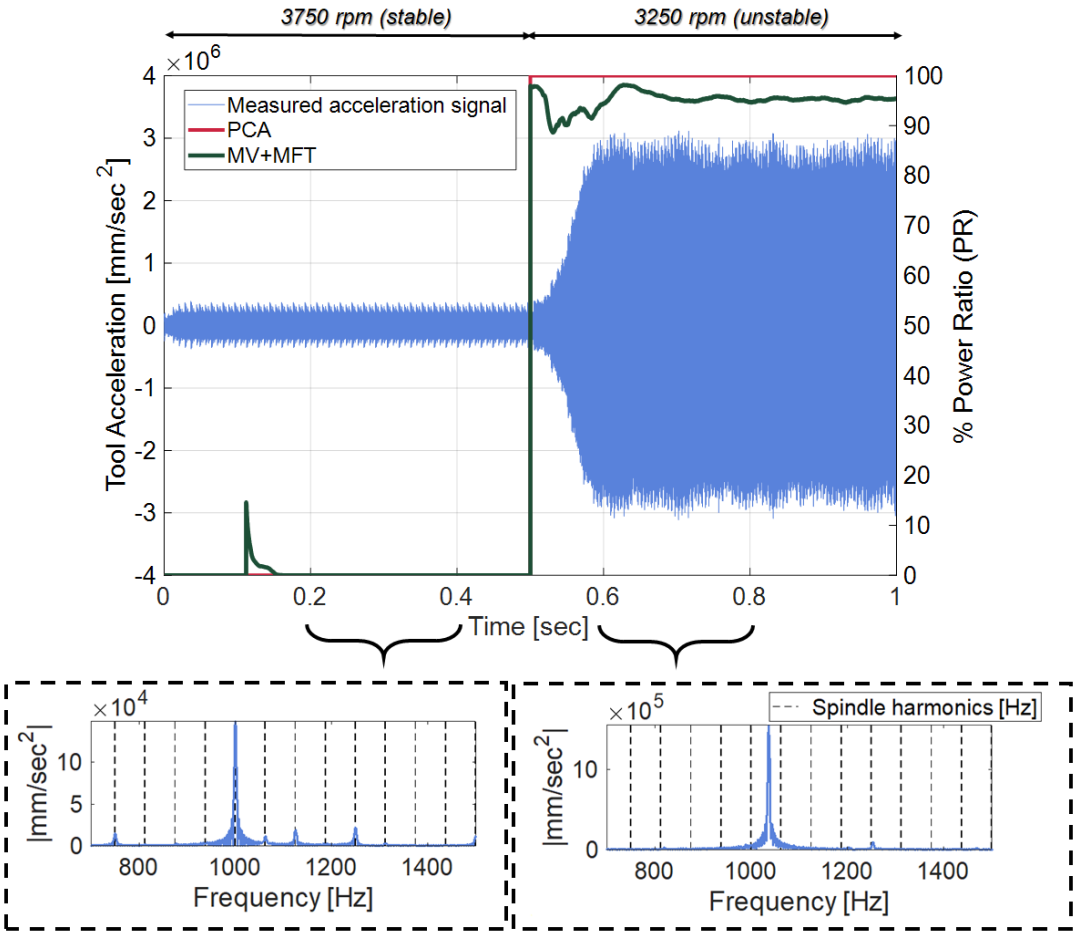

Figure 16: Simulation validation of proposed algorithms, 5 mm depth of cut, 3750 rpm to 3250 rpm

4.5 Experiment Results

Proposed algorithms are also tested on an experimental setup with a flexible fixture/workpiece setup shown in Figure 17. Al6061 material is cut with a 4 flute 10 mm diameter cutter and workpiece acceleration is measured by an accelerometer attached to the fixture. Frequency response of the setup is measured, and it is observed that the fixture exhibits dominant resonances at 60 and 189 Hz as shown in Figure 17. The FRF is obtained to generate stability lobes by impact hammer testing as explained in previous section. As can be seen from Figure 17, the fixture is flexible in X direction which can be interpreted from its FRF as well. The cutting is performed on Y direction.

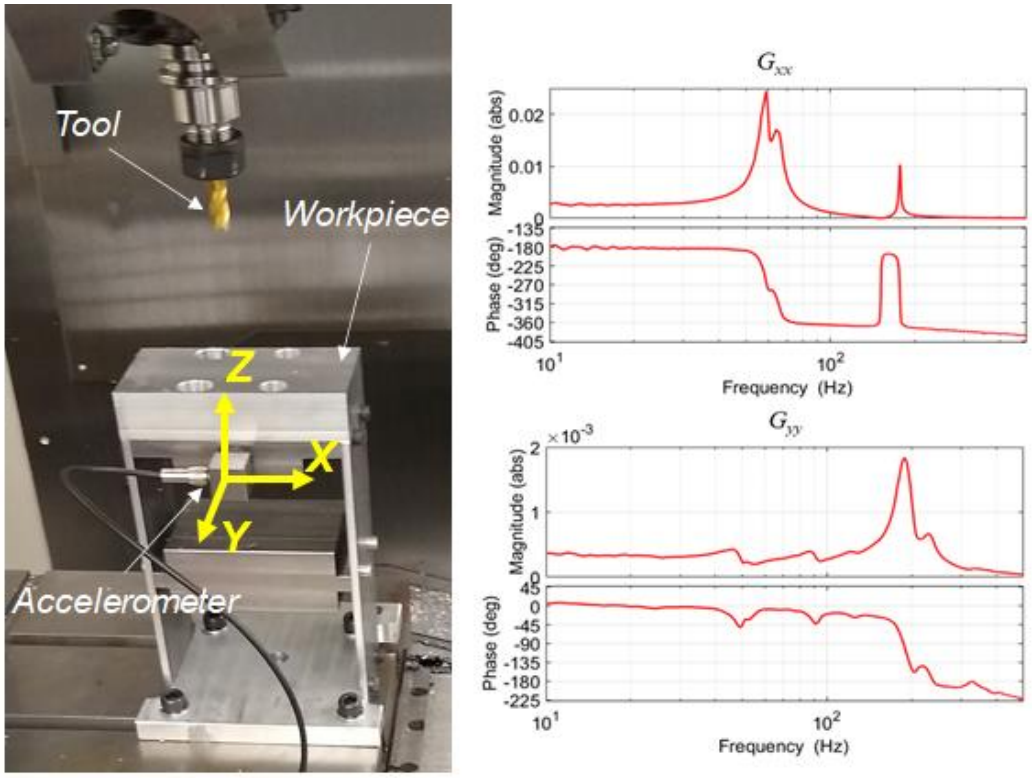

Figure 17: Experimental setup and FRF

Analytical stability lobe diagram of the setup is generated as shown in Figure 18. Cutting (slotting) experiments are conducted at 2000 and 2500 RPM spindle speeds at feedrate of 150 mm/min. Workpiece acceleration is measured at 10 kHz and window size (N) is set to 12 spindle periods for both algorithms, i.e., moving FT and PCA techniques. Hankel matrix is size is selected to be 10 spindle period x 2 spindle period. Power ratio threshold is set to 0.5 for chatter detection which means chatter is detected once total chatter power spectrum is equal to the forced vibration power spectrum. The contribution rate for the PCA technique is set to 0.6 which means 60% of the sum of eigenvalues are considered in the selection step.

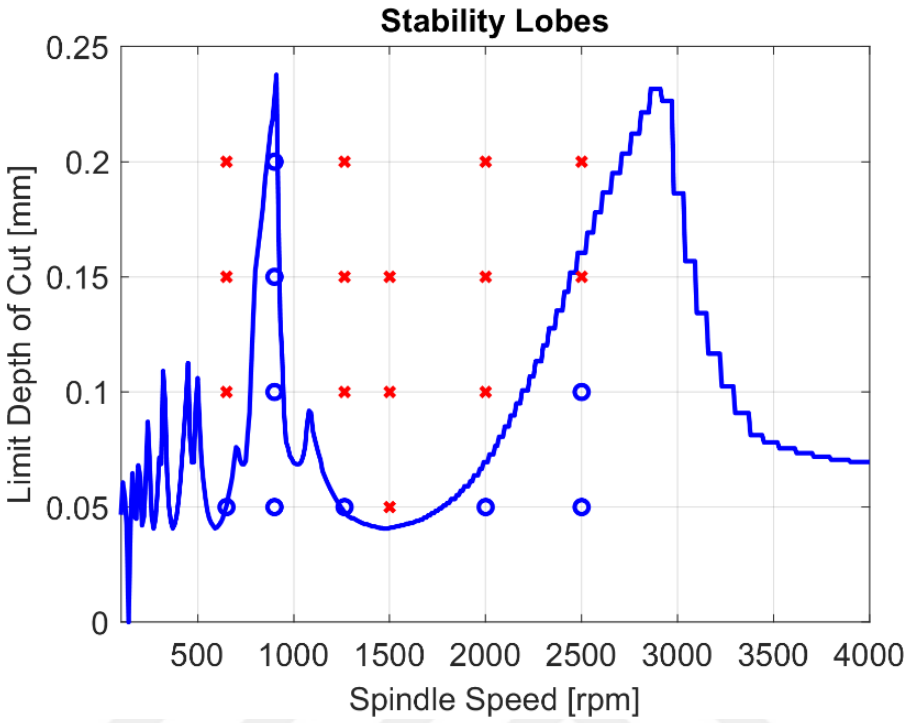

Figure 18: Stability lobe diagram of experimental milling process

Figure 19 shows power ratio of the vibration signal processed by the proposed algorithms for 2500 rpm and 0.1 mm depth of cut. Based on SLD given in Figure 18, this cutting condition is predicted to be stable. However, MV-MFT algorithm indicates several false alarms, i.e., *PR* exceeds the threshold at several instances. This is due to the fact that MV calculation captures total power spectrum of the signal which includes several weak chatter components and noise. However, PCA filters weak components and only considers components once they become dominant enough to be included within the contribution rate, L (see Equation 62). It should be noted that if $L = 1$ would be set, chatter detection would involve those noise and small chatter/forced peaks which would result in false alarms as MV-MFT algorithm does. Thus, it can be claimed that PCA approach is more robust in detecting chatter especially if the machining system is highly flexible like the one used in Figure 17.

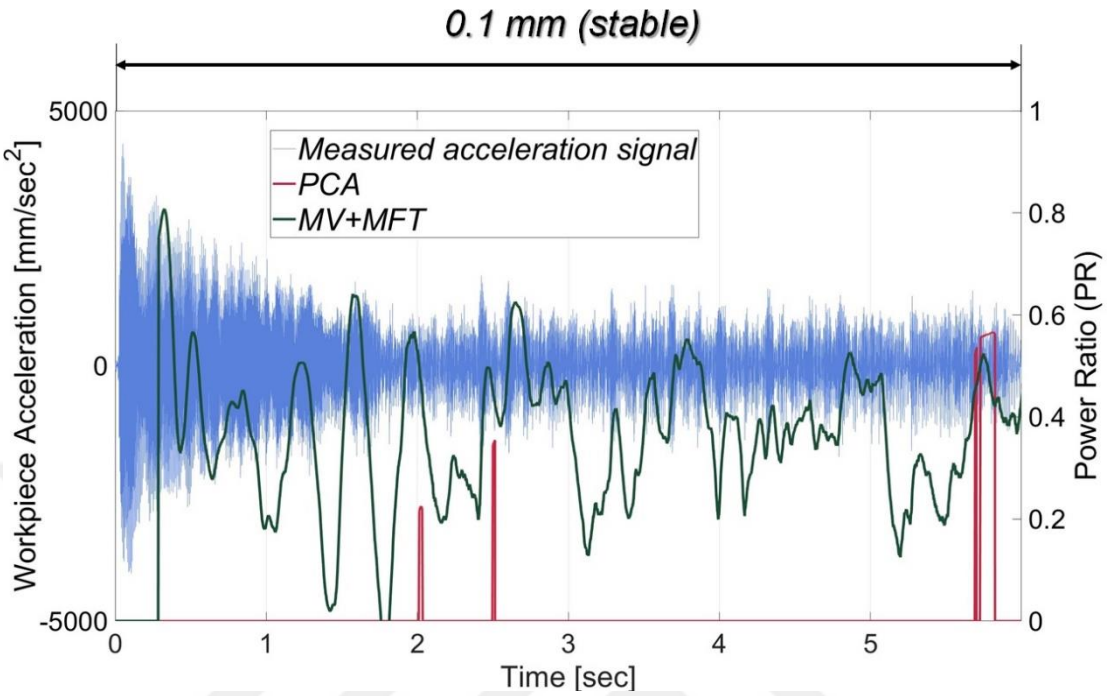

Figure 19: Stable cutting experiment, 2500 rpm

The second slotting experiment is conducted on 2000 rpm with a starting depth of cut of 0.05 mm and a step to 0.2 mm at the end. Note that 0.05 mm depth is stable, and 0.2 mm condition should be chattering. Figure 20 shows vibration signal and the corresponding power ratio. The MV+MFT algorithm produces several false alarms, whereas PCA is much more robust in stable region. Nevertheless, note that when depth of cut is increased to 0.2 mm at around 12th second, chatter rapidly onsets and both algorithms quickly detect increase in the power ratio. Considering the speed of chatter detection, both algorithms behave similarly and detect onset of chattering in a timely manner. However, robustness of the PCA algorithm is visible since it does not provide false alarms and quickly detects chatter once it onsets.

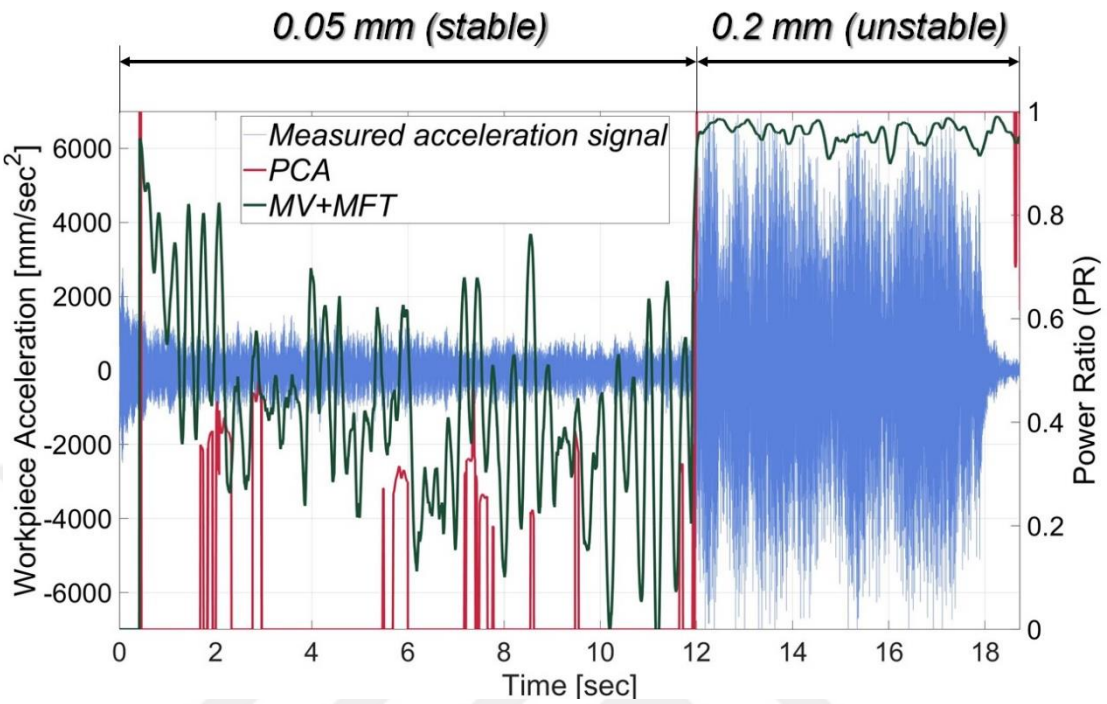

Figure 20: Chatter experiment, 2000 rpm

4.6 Discussion

This section presented two chatter detection algorithms for milling. The first method, MV+MFT, is real-time suitable and capable of detecting chatter once it sets in. However, since it considers total spectral power of the vibration signal, it may be corrupted by noise and other lower power harmonics. As a result, it exhibits false alarms in detecting chatter, which may limit its functionality in a practical setting. In contrast, the PCA based algorithm is more robust in detecting chatter. Although it is computationally more complex than MV-MFT, it can be implemented via recursive algorithms for real-time. It rejects small harmonics signals with small power, and autonomously considers dominant forced and chatter vibration components in the signal. This leads to a more accurate detection of chatter especially when the machining system is flexible. Both algorithms quickly detect onset of chatter. This enables the machining process to be quickly stopped to save the workpiece, or the spindle speed can be changed to stabilize the process.

5 Adaptive Spindle Speed Optimization

5.1 Introduction

Mitigation of chatter vibrations is up most critical for all the machining processes [2,22,31]. Stability lobe diagrams are used to select spindle speed – depth of cut parameters for avoiding chatter vibrations. Even though chatter vibrations can be avoided by selecting process parameters underneath the stability lobes, forced vibrations can still deteriorate surface quality in ultra-precision machining applications or during machining of highly flexible workpieces such as impeller blades or monolithic components used in aerospace industries. This section presents an optimal spindle speed selection technique to minimize the “total” vibration level in milling processes.

This section presents two extremum seeking control based adaptive spindle speed selection algorithm that searches for the optimal spindle speed, which minimizes overall vibration level during a milling process. Firstly, extremum seeking control is explained. Two methods based on extremum seeking control are explained with their stability analysis, and how they are used in milling vibration reduction is presented. Both methods are performed on time domain milling simulation, and results are discussed.

5.2 Adaptive Spindle Speed Seeking Algorithm

As presented in previous sections, stability and the level of vibrations in the milling process can be predicted from Equation 31. However, such analytical stability prediction requires various process parameters such as the tool geometry, material properties and an accurate characterization of the structural dynamics [4]. Most of these parameters may contain certain level of uncertainty, which degrades accuracy of the stability lobe diagrams. Additionally, the linearized model does not include non-linearities such as the effect of process damping [49], tool fly-over [50,51], wear and cutter runout [52], which are frequently observed in practice and significantly alter the

stability lobe diagrams. Lastly, although selecting a spindle speed – depth of cut pair under the stability limits should mitigate occurrence of chatter vibrations, it does not necessarily minimize adverse effects of forced vibrations. Selecting the most optimal spindle speed that also minimizes the forced vibrations and generates a smooth surface finish is a practical challenge.

This section presents a real-time optimization method of spindle speed in milling process. The proposed algorithm bases on the spindle speed variation (SSV) approach developed for suppressing chatter [53]; however, as opposed to directly disturbing regeneration effect and suppressing chatter, here it is used to determine the optimal spindle speed in a SLD and minimize overall tool vibrations. The algorithm is originally based on the sinusoidal extremum seeking control (ESC) technique used in on-the-fly optimizing linear/non-linear systems.

5.3 Extremum Seeking Control

Extremum seeking control is used to optimize a performance measure in terms of the steady state output of the system [54,55]. It is a real time adaptive control approach which includes adaptation of the system parameters to obtain optimum system output [54,55]. It is used for both linear and nonlinear systems [55], dynamical systems with time varying parameters [55].

In this thesis, the conventional extremum seeking control-based spindle speed optimization is first explained, and its validation on simulation is presented. Secondly, dither-free least squares-based extremum seeking control scheme is examined and its implementation on spindle speed optimization is detailed.

5.3.1 Conventional Extremum Seeking Control Based-Spindle Speed Optimization

Figure 21 shows generalized basic extremum seeking control block diagram. As can be seen, a performance measure of an unknown system is optimized in real time. It is based on perturb and observe approach. In other words, system parameters are changed, and a performance measure (cost function to minimize or objective function to maximize) is defined based on system outputs. Figure 22 shows an example static cost function to minimize with single global optimum point. Here, the simplest case is shown where the system dynamics respond to system parameter instantaneously. In the Figure 22, u^* represents optimum point for system parameter or input which is adjusted to optimize performance measure J to its optimum point J^*.

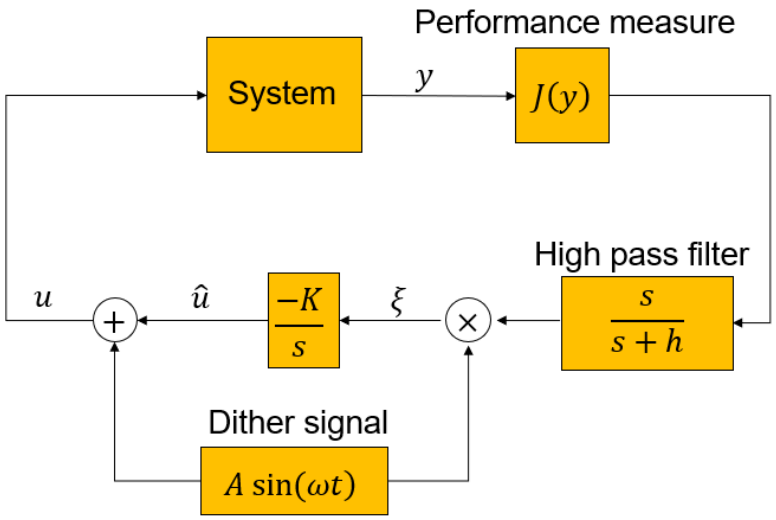

Figure 21: Conventional extremum seeking control block diagram

Consider the point u_1 in the Figure 22. As shown in Figure 21, a sinusoidal perturbation is added to the system parameter u_1 which results in sinusoidal output for the performance measure J_1. It can be interpreted that if the system parameter is increased, the cost decreases, or the cost increase if the system parameter is decreased. If sinusoidal signals of both system parameter and cost values is zero mean, the multiplication of those (ξ) yields mostly negative signal. For the point u_2 and J_2, it yields mostly positive signals. Therefore, as shown in Figure 21, by integrating this demodulated signal with a negative gain, optimization converges to optimum points.

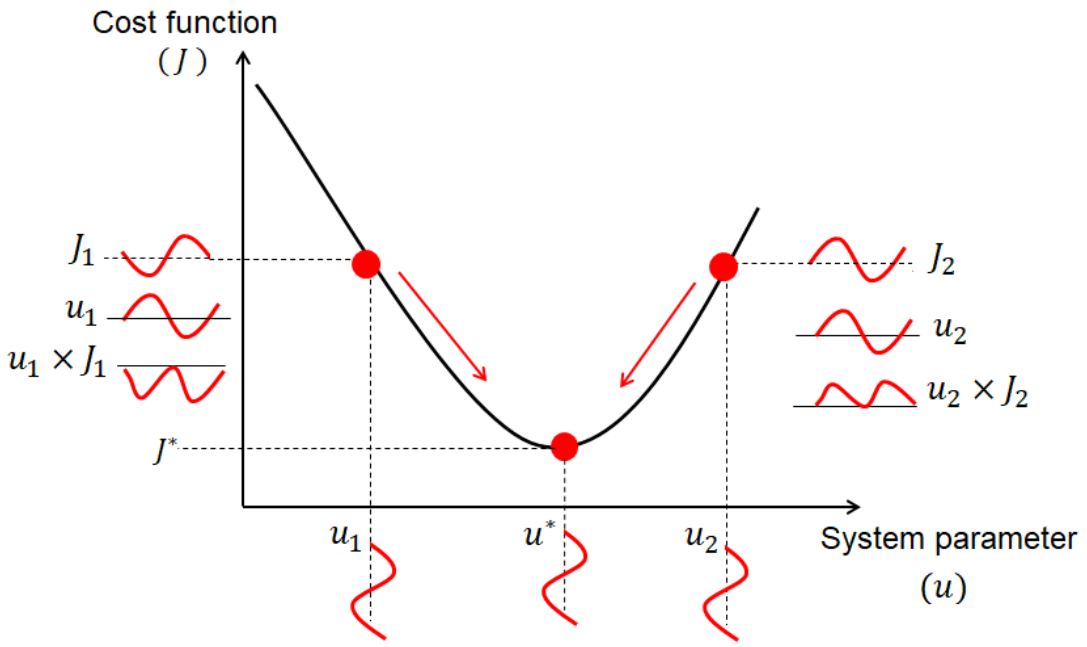

Figure 22: Representation of static convex cost function map

For the stability analysis of this static map, consider the twice differentiable static convex cost function (performance) map given in Figure 22. It should be noted that minimization is the case for vibrations in milling as presented in chapter 3. The performance measure can be postulated as follows [55]:

$$J(\theta) = J^* + \frac{J''}{2}(u - u^*)^2 \tag{64}$$

where J is the performance measure, u is the plant input, superscript $*$ denotes the extremum of the map. For maximization, sign of K in Figure 21 is replaced with its negative. The algorithm converges to the optimal performance measure J^* by reducing the difference $u - u^*$. The dither signal enables to predict the gradient of the performance map. If the dither signal is substituted into Equation 64,

$$J(\theta) = J^* + \frac{J''}{2}\left(\tilde{u} - Asin(\omega t)\right)^2 \tag{65}$$

where $\tilde{u}$ is the estimation error given by $\tilde{u} = u^* - \hat{u} = u^* - u + Asin(\omega t)$. Equation 65 is further expressed by including trigonometric functions as below.

$$J(\theta) = J^* + \frac{J''}{2}\tilde{u}^2 - AJ''\tilde{u}\,sin(\omega t) + \frac{J''}{4}A^2(1 + cos(2\omega t)) \quad (66)$$

By adding high-pass filter effect, the DC term of J^* is removed from Equation 66. The demodulation signal ξ is then obtained as follows:

$$\left.\begin{aligned} \xi \approx \frac{J''}{2}\tilde{u}^2 Asin(\omega t) - A^2 J''\tilde{u}\,sin^2(\omega t) + \frac{J''}{4}A^3 sin(\omega t)\,cos(2\omega t) \approx \\ \frac{J''}{2}\tilde{u}^2 Asin(\omega t) - \frac{\tilde{u}A^2J''}{2} + \frac{\tilde{u}A^2J''}{2}cos\,(2\omega t) + \frac{A^3J''}{8}\big(sin(3\omega t) - sin(\omega t)\big) \end{aligned}\right\} \quad (67)$$

Since the u^* is constant, time derivative of estimation error becomes $-\dot{\hat{u}}$. For the local analysis, estimation error becomes:

$$\begin{aligned} \tilde{u} = \frac{K}{s}\Bigg[\frac{J''}{2}\tilde{u}^2 Asin(\omega t) - \frac{\tilde{u}A^2J''}{2} + \frac{\tilde{u}A^2J''}{2}cos\,(2\omega t) \\ + \frac{A^3J''}{8}\big(sin(3\omega t) - sin(\omega t)\big)\Bigg] \end{aligned} \quad (68)$$

Here, high frequency terms are neglected since they are attenuated with the integrator, and the $\tilde{u}^2$ term is neglected since interest is on local analysis.

$$\left.\begin{aligned} \tilde{u} = \frac{K}{s}\left[-\frac{\tilde{u}A^2J''}{2}\right] \\ \dot{\tilde{u}} \approx -\frac{KA^2J''}{2}\tilde{u} \end{aligned}\right\} \quad (69)$$

Equation 69 is stable since $KJ'' > 0$ (convex function assumption). Therefore, ũ converges to 0, i.e., $\hat{u}$ converges to u^*. It is concluded that, for a static performance map, method converges to the "locally" optimum point provided the perturbation frequency ω is sufficiently large [55]. [55] provides more detailed stability analysis for multi parameters, plant with dynamics, linear time-varying stability, linear time-invariant stability.

In this section, conventional extremum seeking control-based optimization is applied to mitigate total vibrations in the milling process. Block diagram of the proposed approach is given in Figure 23.

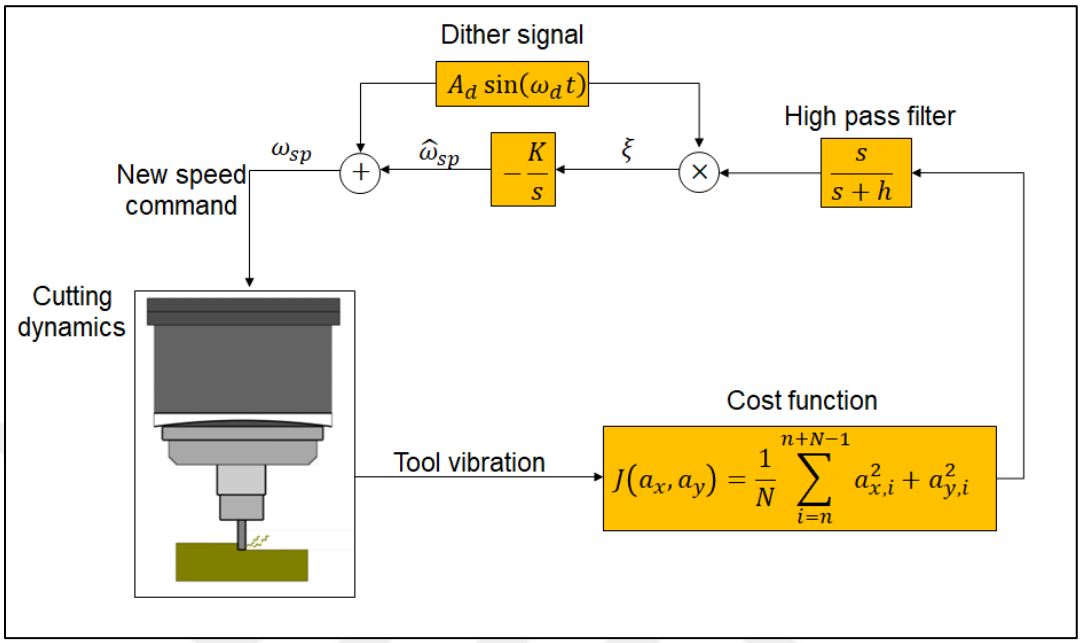

Figure 23: Conventional extremum seeking control-based spindle speed optimization block diagram

As shown, a sinusoidal dither signal, i.e., a spindle speed variation (SSV), is sent to the spindle speed controller, and the response of the process is evaluated via a cost function defined as:

$$J[k] = \frac{1}{N} \sum_{i=k-N-1}^{k} a_x^2[i] + a_y^2[i] \qquad 70$$

Here, J contains total machining vibrations in X-Y directions accumulated over a time duration of last spindle revolution where N is the total number of samples of last spindle revolution. Next, the cost function is high pass filtered to remove its average (DC) value and modulated (multiplied) by the commanded spindle speed variation signal. Notice that if the cost function follows a sinusoidal trend in phase with the spindle speed variation command (dither signal), the integration of ξ signal provides a positive spindle speed increment $\Delta\omega_{sp}$ whereas if it is out of phase, it generates a negative value. This is repeated for each time sample and spindle speed is updated. Since the cost

function is based on the total vibrations, this optimization algorithm converges to an operating point, e.g., a spindle speed, which minimizes overall machining vibrations.

The proposed algorithm does not require prior knowledge of the machining dynamics. Key tuning parameters are the integrator gain K, high pass filter frequency ω_{hpf}, sinusoidal dither (spindle speed variation) frequency and its amplitude. The integrator gain controls the step size and should be selected small enough to ensure stable convergence, and a smooth spindle speed command. The dither signal amplitude should be selected so that it disturbs and induces a change in the machining vibration level. In practice 20-50 [rpm] would be sufficient. The dither signal period should be selected sufficiently small to ensure velocity controller of the spindle is able to follow the commands in real time. Lastly, the high pass filter frequency needs to be selected well below the dither frequency to ensure that while changes in the process are detected, the DC (average) vibration value is rejected.

Proposed method is validated by using the time domain milling simulations. Figure 24 shows the forced vibration (static component of the semi-discrete analysis) reduction by proposed method. Figure 24a shows SLD (red curve), singular values of DC component (black curve), and the initial (I_1, I_2) and final (F_1, F_2) points of the simulation. Time domain simulation results are shown in Figure 24b where blue is normal (Y) direction acceleration, green is feed (X) direction acceleration, red is cost (defined in Equation 70), and black is spindle speed over time. As can be seen, both conditions converge to the minimum-singular values point on the map. This is because the cutting condition is asymptotically stable at the depth of cut so that only forced vibrations are experienced during cutting. Once the spindle speed adaptation is converged, total vibration level is reduced by ½. Note that effect of the spindle speed controller dynamics is not considered in the simulation, which brings a low-pass effect to the spindle speed change between commanded and real speeds.

Figure 25 shows when the process is started at an unstable cutting – chattering – condition. As shown in Figure 25, simulation starts outside of the SLD, and the

proposed algorithm optimizes the spindle speed to a stable one. Reduction on the vibration in this case is much more than the previous simulation since it stabilizes the machining system. First rapid change is dependent on the vibration level at that time and gain of the optimization. It can be reduced by ensuring small integrator gain.

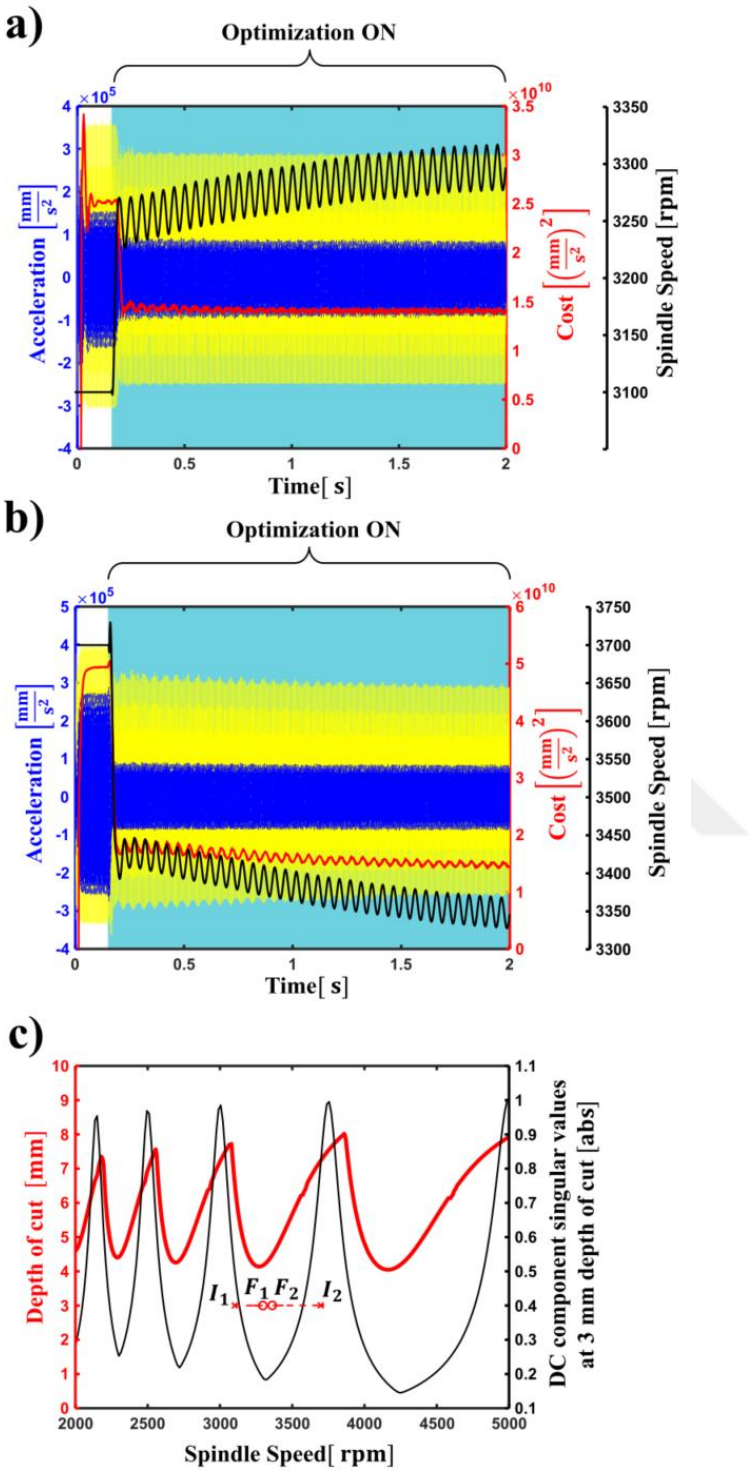

Figure 24: Forced vibration reduction by conventional ESC-based spindle speed optimization method , $A_d = 20$, $\omega_d = 20\ [Hz]$, $K = 3 \times 10^{-8}$, $\omega_{hpf} = 4\ [Hz]$, time domain results a) $I_1 \rightarrow F_1$, b) $I_2 \rightarrow F_2$, c) stability lobes & DC component singular values and initial (I_1, I_2) and final (F_1, F_2) conditions before & after optimization, optimizations are started at 0.175s and 0.145s respectively (at 10th spindle revolution after simulation starts), yellow: feed, blue: normal direction acceleration

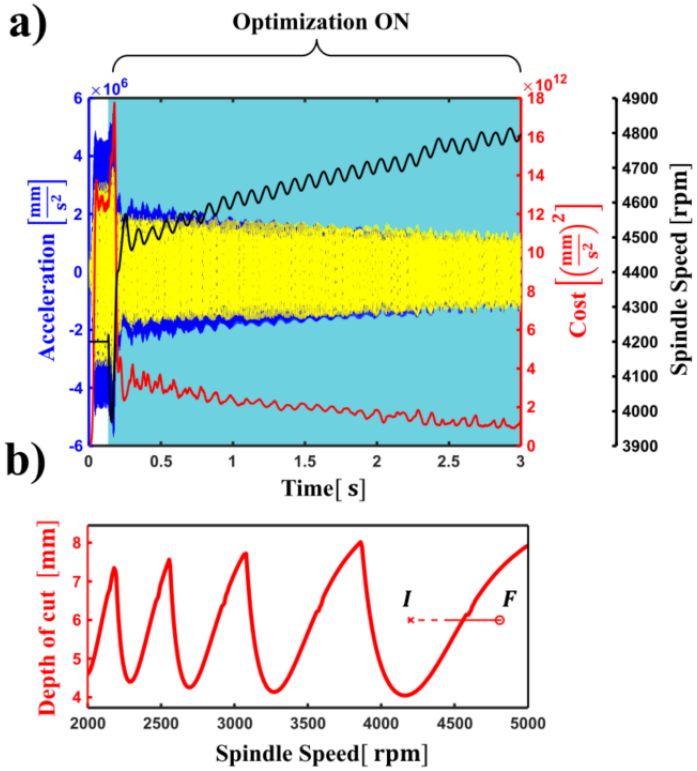

Figure 25: Chatter stabilization by conventional ESC-based spindle speed optimization method , $A_d = 20$, $\omega_d = 10\ [Hz]$, $K = 8 \times 10^{-11}$, $\omega_{hpf} = 2\ [Hz]$, time domain results a) $I \rightarrow F$, b) stability lobes & DC component singular values and initial (I) and final (F) conditions before & after optimization, optimization is started at 0.26s (at 10th spindle revolution after simulation starts), yellow: feed, blue: normal direction acceleration

5.3.2 Dither-free Extremum Seeking Control-Based Spindle Speed Optimization

Previous section presented fully real-time conventional extremum seeking control-based spindle speed optimization on simulation. As can be seen, it optimizes successfully on the time domain simulation which do not include spindle speed controller dynamics. However, spindle speed controller of actual machine may not be able to follow produced speed commands on the simulation. This is because actual machine speed controller will have a low-pass effect, i.e., dynamics to follow speed commands. Therefore, cost calculation and speed command generation are adjusted in this section. Instead of conventional extremum seeking control, dither-free least squares-based extremum seeking control is more appropriate to proposed method.

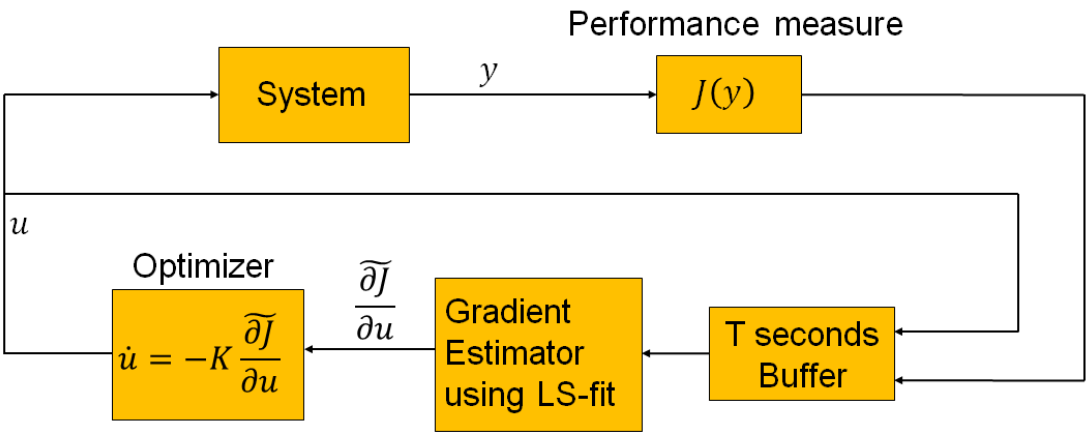

Figure 26: Dither-free least squares-based extremum seeking control block diagram

Hunnekens proposed dither-free least-squares-based extremum seeking control scheme [56]. Here, instead of adding dither signal, gradient information of the cost function map is estimated by buffering cost and system input values and applying first order least squares. Figure 26 shows the block diagram of the approach. As can be seen, last T seconds data is buffered and used to estimate the gradient using a first order least squares fit $au + b$. The convex minimization problem becomes as follows [56]:

$$\min_{a,b} \int_{-T}^{0} \Big(J\big(u(t+\tau)\big) - \big(au(t+\tau) + b\big) \Big)^2 d\tau \qquad 71$$

Slope of the regressed line is considered as the estimated gradient information. Although the explicit solution for the estimated gradient can be computed, it can be solved numerically as well. Here, adaptation gain K should be selected small enough to ensure the separation of time scales of optimization and plant dynamics [56]. Stability analysis for this method is given in [56] where they use assumption of static map, non-zero initial value, and Lyapunov-Razumikhin theorem to prove stability. It is shown that, for sufficiently small KT multiplication, a small neighborhood of $u(t)$ is used to obtain the estimation of gradient. Besides, based on Lyapunov theorem, it is shown that for a small enough KT, $\lim_{t\to\infty} u(t) = u^*$ [56].

For the spindle speed optimization, this method is adjusted where cost is total vibration and system input is spindle speed commands. Figure 27 shows block diagram of the proposed approach. As shown, cost function and the corresponding spindle speeds are

buffered. By first order fit with least-squares method, gradient of the cost function-spindle speed map is estimated. This estimation is used to integrate the spindle speed. The details of the algorithm are given below.

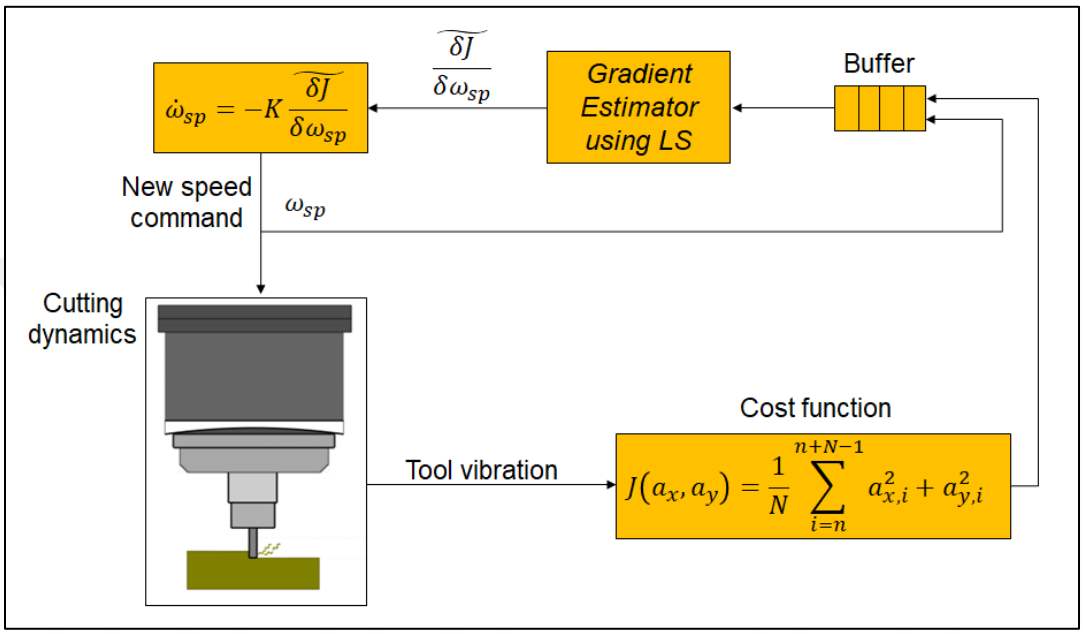

Figure 27: Dither-free least squares-based spindle speed optimization

Firstly, cost function is defined as total vibration which is the same as previous section. However, calculation of cost function is adjusted here. The cost is given as:

$$J(a_x, a_y) = \frac{1}{N}\sum_{i=n}^{n+N-1} a_{x,i}^2 + a_{y,i}^2 \qquad 72$$

where a_x and a_y are acceleration of feed (X) and normal (Y) directions, N is the number of samples within 1 spindle revolution. The spindle speed is adjusted at the beginning of every N_c revolutions. Since the cost value contains total acceleration accumulated over only last spindle revolution, spindle speed is kept constant over the rest of $N_c - 1$ revolutions. Length of N_c is adjusted based on spindle speed controller dynamics. In other words, it is ensured that correct spindle speed and cost value is matched to perform optimization. For this purpose, N_c is arranged to ensure commanded spindle speed is reached, and acceleration is at its steady state region.

Since there is not any update or calculation during the rest $N_c - 1$ revolutions, by adjusting its size, enough time can be left for spindle speed controller to reach generated speed commands. After that, N_B number of spindle speed – cost value pairs are collected in a buffer which has first-in first-out logic. It should be noted that buffer contains N_B number of speed-cost pairs of every N_c revolution. In other words, each sample in the buffer is valid for N_c revolutions. Therefore, actual buffer size becomes $N_B \times N_c$ revolutions. Once the buffer is filled, least squares fit is applied to spindle speeds and cost values to estimate the gradient of the cost function map. Slope of the fitting is used as the estimated gradient, and it is used to integrate the spindle speed to its new value. Positive slope means cost value increases if the spindle speed is increased. Therefore, spindle speed should be decreased for positive slope since minimization of acceleration is performed. This process is repeated until spindle speed converges to an optimized value. For better understanding, steps for cost function calculation and buffer filling, spindle speed – cost value matching, fitting, and updating the spindle speed is detailed as follows.

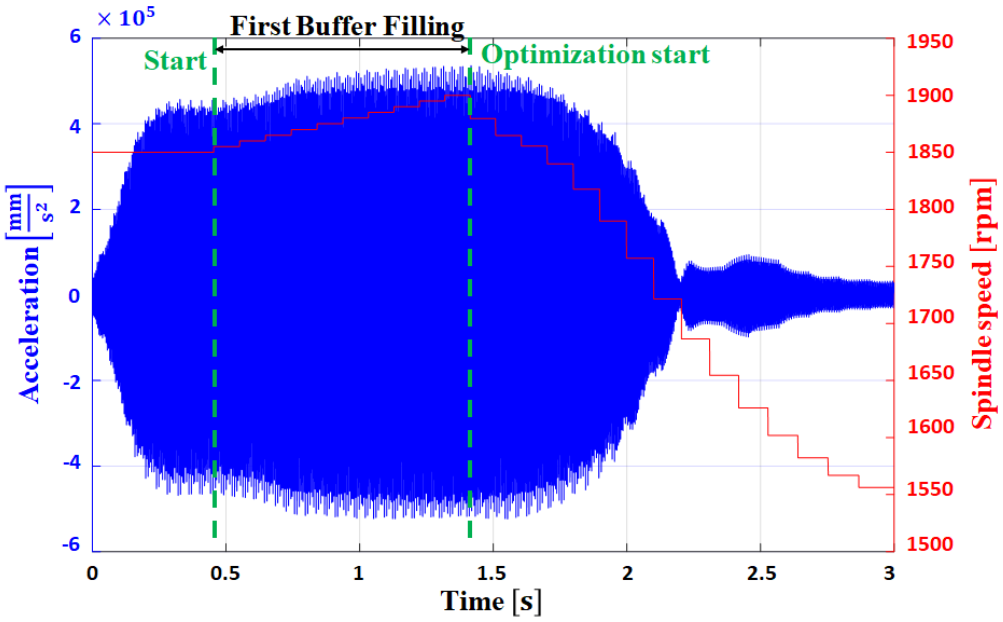

Figure 28: Dither-free spindle speed optimization understanding: first buffer filling

As can be seen in Figure 28, algorithm is started at around 0.45 s (first green dotted line). First buffer of N_B data points is filled by constantly increasing or decreasing

(increasing in this example) the spindle speed at every N_c revolutions. Constantly increasing or decreasing for the first buffer is performed in order not to cause matching a spindle speed to several cost values (having infinity slopes). Buffer consists of corresponding spindle speed, and the cost value accumulated over a spindle revolution with that speed at every N_c revolutions. First buffer filling region can be seen detailed (zoomed) in Figure 29. It should be noted that, right Y axis in Figure 28 is spindle speed where it is corresponding cost value in Figure 29. As shown in Figure 29, after start of the first buffer filling region, cost values are calculated at the beginning of every N_c revolutions (which is 3 in this example). Calculated cost values are matched with their corresponding spindle speeds as shown in Figure 30. It can also be seen that a spindle speed is kept constant within N_c revolutions, and the cost value is calculated at the end of every N_c revolutions. Corresponding spindle speeds and cost values are matched to fill the buffer. Once the buffer (N_B cost-spindle speed pairs) is filled, optimization process starts. Here, N_B is another time scale where each element in the buffer represents cost values calculated at every N_c revolutions and their corresponding spindle speeds. In other words, actual buffer size is $N_B \times N_c$ revolutions. For instance, in the example given in Figure 28, buffer size (N_B) is 10 where each element in the buffer is calculated over 3 revolutions (N_c). Therefore, actual buffer size is 30 revolutions.

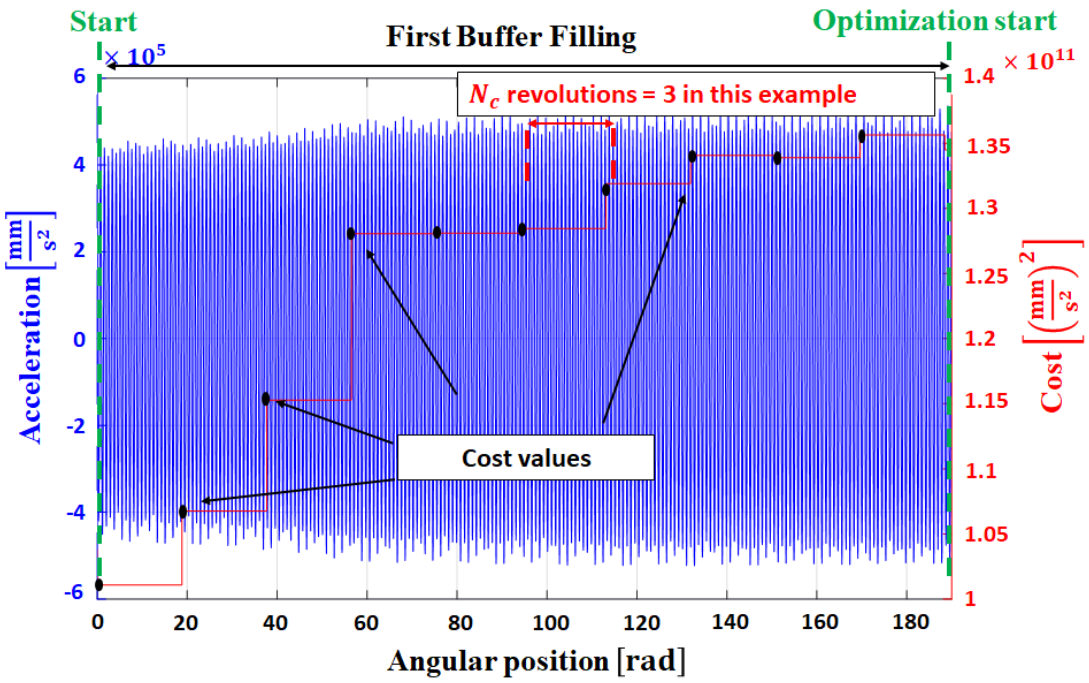

Figure 29: Dither-free spindle speed optimization understanding: first buffer filling detailed

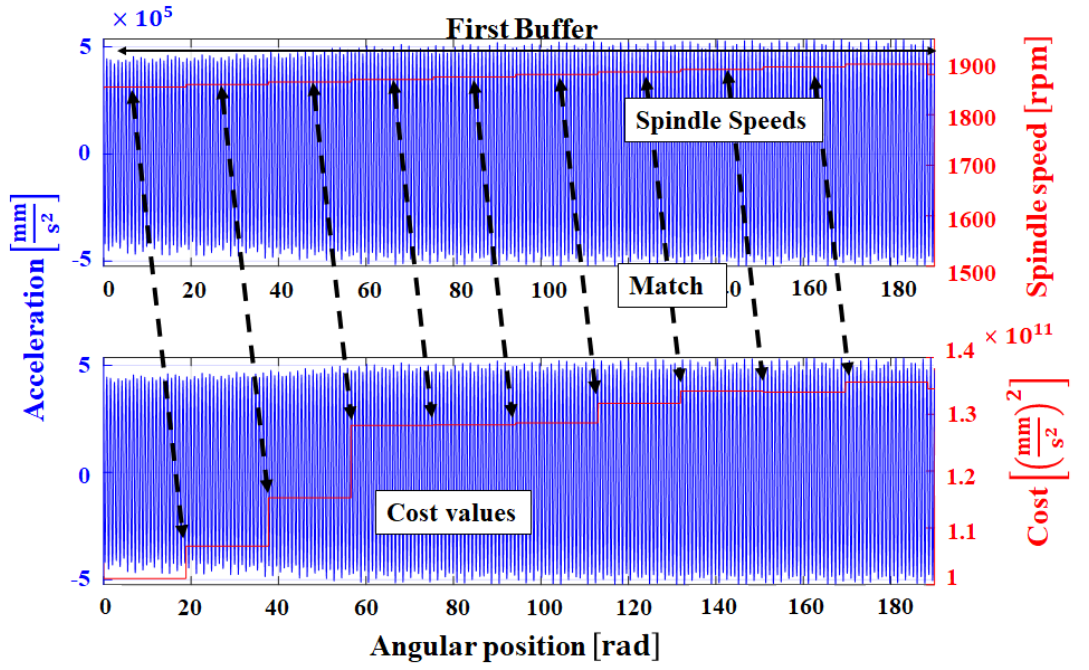

Figure 30: Dither-free spindle speed optimization understanding: spindle speed - cost value matching

Once the matching is performed and first buffer is filled, a first order least squares fitting is applied to spindle speed – cost values as shown in Figure 31. Slope of the fitting is used as the estimated gradient.

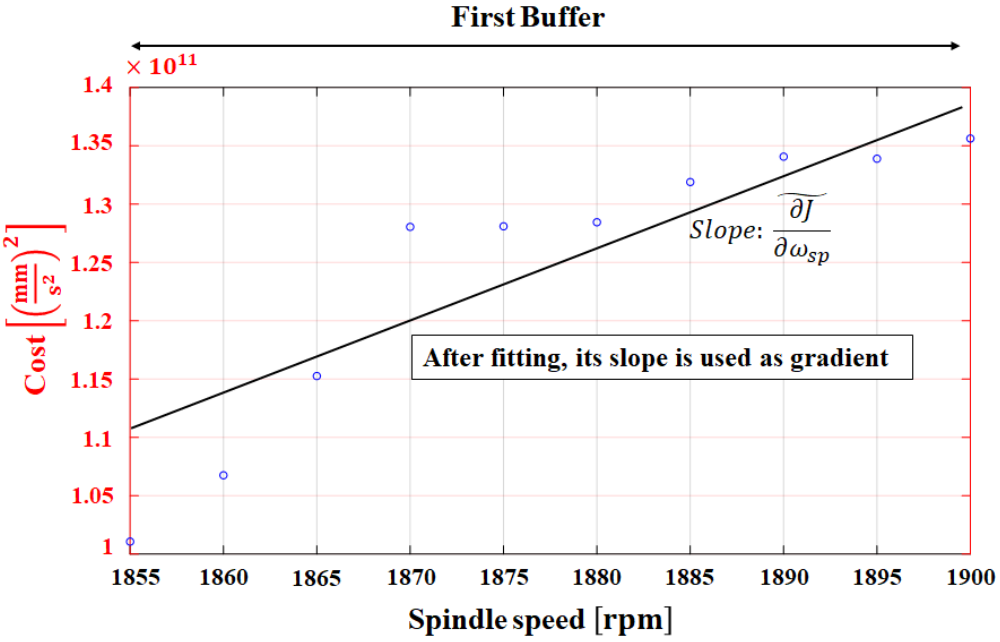

Figure 31: Dither-free spindle speed optimization understanding: fitting

After estimating the gradient, spindle speed is updated as given in Figure 32. The update rule is as below:

$$\omega_{sp_{new}} = \omega_{sp_{old}} - K\frac{\widetilde{\partial J}}{\partial \omega_{sp}} T_B \qquad 73$$

where $\omega_{sp_{old}}$ is previous spindle speed, i.e., last spindle speed in the buffer, $\omega_{sp_{new}}$ is updated spindle speed, K is integrator gain, $\widetilde{\partial J/\partial \omega_{sp}}$ is estimated gradient, T_B is the time duration passed over previous buffer, i.e., time duration length of previous buffer ($N_B \times N_c$ revolutions). New spindle speed is kept constant for N_c revolutions and its corresponding cost value is calculated. The buffer is moved 1 sample with first in first out logic. In other words, first spindle speed – cost value pair of the buffer is deleted, and new spindle speed – cost value pair is added. As can be interpreted from Figure 31, if the slope (estimated gradient) is positive, i.e., if the cost value is increasing by increased spindle speed, update rule decreases the spindle speed. Also, while

approaching to an optimal point, the slope will be much lower than the beginning and will yield convergence to the optimum spindle speed.

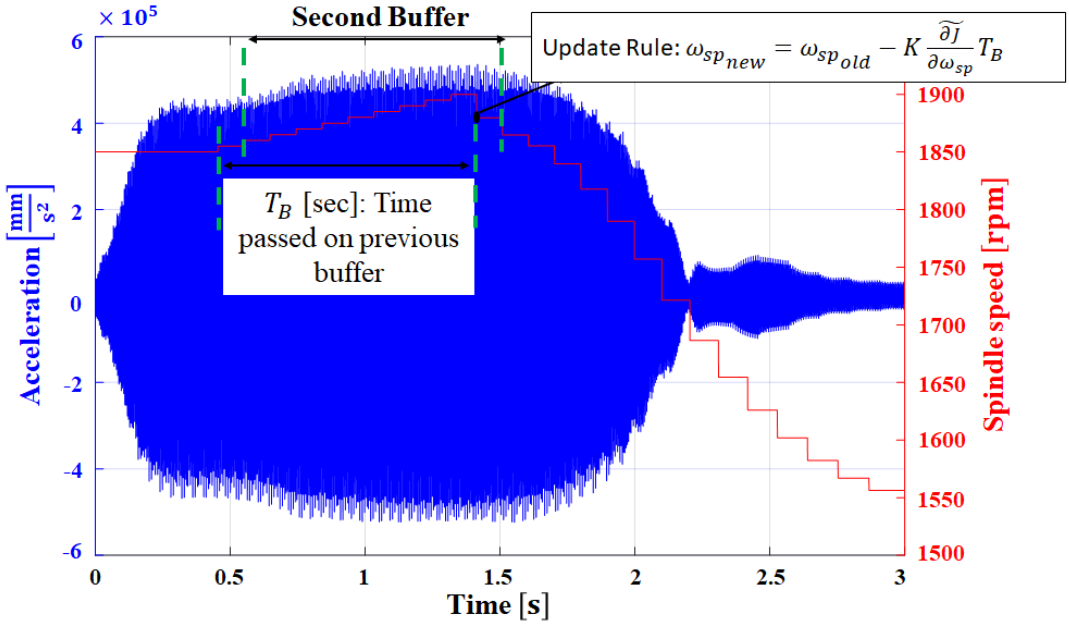

Figure 32: Dither-free spindle speed optimization understanding: spindle speed updating

Proposed method is validated by using the time domain milling simulation. Figure 33 shows when the process is started at a stable cutting condition, i.e., forced vibration (static component of the semi discrete analysis) reduction. The machining condition is the same as given in Figure 25. Figure 33a and Figure 33b show time domain simulation results where blue is normal (Y) direction acceleration, yellow is feed (X) direction acceleration, red is cost value (defined in Equation 72), and black is spindle speed over time. Figure 33c shows SLD (red curve), singular values of DC component at 3 mm depth of cut (black curve), and initial points (I_1, I_2) of spindle speeds – depth of cuts, and final (F_1, F_2) points after optimization converges. First buffer is filled with decreasing spindle speed by 3 rpm at every $N_c = 5$ revolutions. Once the buffer ($N_B =$ 10 data points of spindle speed – depth of cut pairs) is filled, first update of optimization is started. As shown both conditions converge to the minimum singular values point on the map. This is because cutting condition is asymptotically stable at the depth of cut of 3 mm, i.e., only forced vibration are experienced during cutting. Once the spindle speed adaptation is converged, vibration level in normal direction is reduced by ½. It

can also be seen that first rapid increase of spindle speed in conventional method is decreased with stepwise changes of spindle speeds. Besides, since the algorithm presented in previous section was generating updated spindle speed commands at every time sample, actual speed controller dynamics would not be able to reach the commanded speed. The delay between those would have negative effect on actual machining tests. This problem can be solved by adjusting N_c in dither-free algorithm. Here, thanks to stepwise increase of spindle speed, enough time for spindle speed controller to reach generated spindle command can be left by adjusting N_c.

Figure 34 shows when the process is started at an unstable cutting – chattering – condition. As can be seen, initial machining condition (*I*) is at above the stability lobes, and the proposed algorithm optimizes the spindle speed to a stable condition. Reduction on the vibration in this case is also much more than only reducing forced vibrations since the process is stabilized from chatter vibrations. First buffer is filled with decreasing spindle speed by 3 rpm at every 5 revolutions (N_c=*5*). Optimization update starts after 10 data points of speed – cost pairs are collected (N_B=*10*). As shown, thanks to stepwise increase/decrease of spindle speed based on N_c, enough time for spindle speed controller to reach generated spindle command can be left in milling experiment.

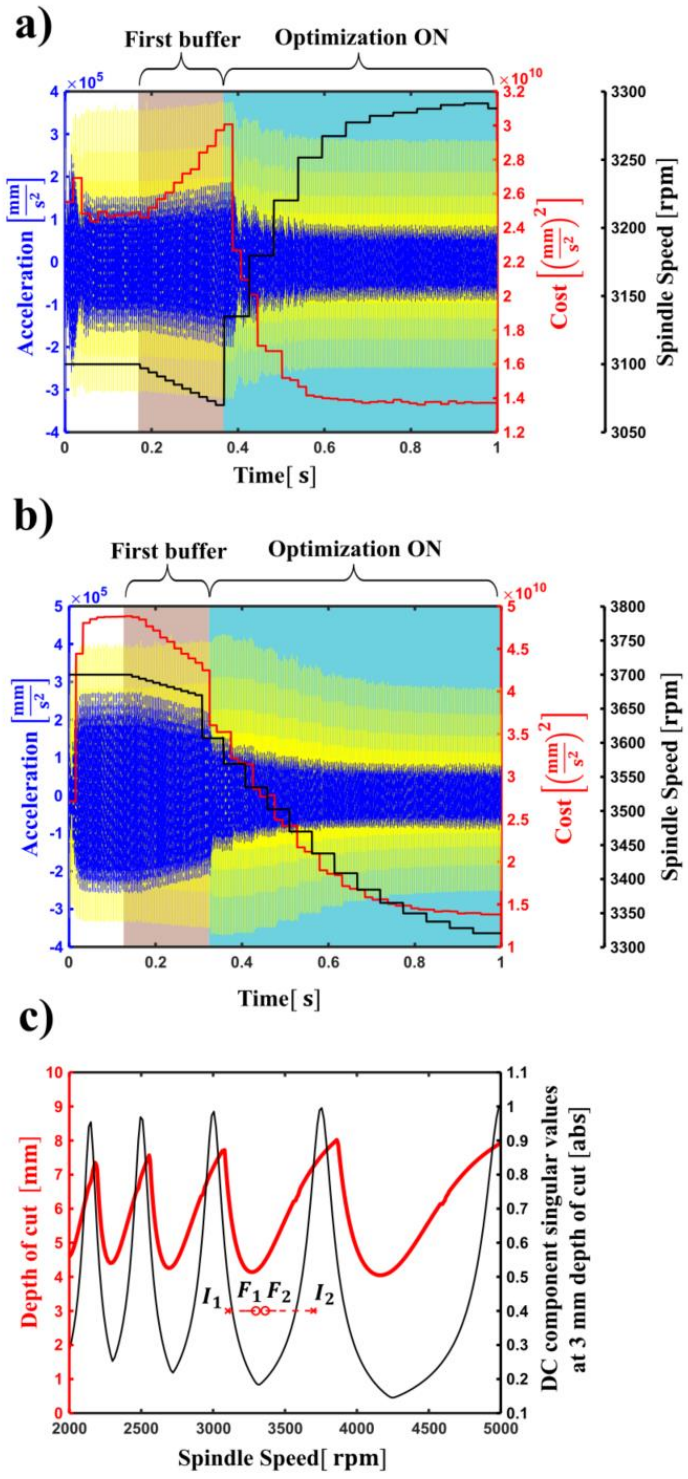

Figure 33: Forced vibration reduction by proposed spindle speed optimization method, N_c = 3 revolutions, N_B = 10 speed-cost pairs, $K = 6 \times 10^{-6}$, time domain results a) $I_1 \rightarrow F_1$, b) $I_2 \rightarrow F_2$, c) stability lobes & DC component singular values and initial (I_1, I_2) and final (F_1, F_2) conditions before & after optimization. First buffer filling is started at 0.175 and 0.145 s respectively (at 10th spindle revolution after simulation starts)

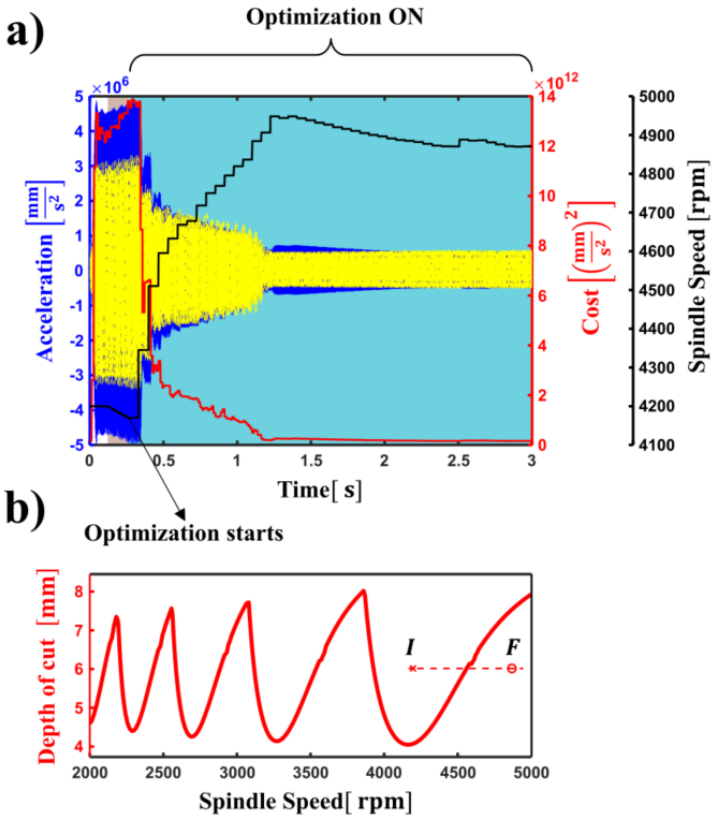

Figure 34: Chatter stabilization by proposed spindle speed optimization method, N_c = 5 revolutions, N_B = 10 speed-cost pairs, $K = 6 \times 10^{-8}$, a) $I \rightarrow F$ time domain results, b) stability lobes and initial (I) and final (F) conditions before & after optimization. First buffer filling is started at 0.128 s (at 10th spindle revolution after simulation starts)

6 Conclusion

This thesis presents dynamic milling process modeling, vibration analysis, chatter vibration detection, and vibration mitigation through spindle speed optimization in the milling process. Two different online chatter detection methods are presented. Although the lightweight algorithm is suitable for real-time processing, it generates false alarms if the tool or workpiece is too flexible or if vibration sensing is noisy. This issue is addressed and solved with a robust algorithm, which sacrifices computational simplicity. Both algorithms detect chatter promptly. Furthermore, adaptive spindle speed optimization is examined. A time domain milling simulation, which enables spindle speed changes at any time step, is presented. More detailed vibration analysis based on the semi-discrete method is performed and presented in time domain simulations. An adaptive optimization method, conventional extremum seeking control-based spindle speed optimization, is presented. Although it is shown to mitigate total vibration in the process, the method may generate rapid spindle speed changes. Since actual spindle speed controller dynamics may not allow reaching fast speed changes at every time step, it causes an incorrect estimation of the spindle speed – cost function map gradient. On the other hand, a dither-free extremum seeking control-based optimization method is presented, where enough time can be left to reach commanded speed by adjusting spindle speed with stepwise changes at every predetermined number of revolutions. Both methods are presented and validated by time domain simulations. Results show that not only chatter vibrations but also forced vibrations are reduced by adaptive spindle speed optimization methods.

Bibliography

[1] Schmitz, T. L., and Smith, K. S., 2009, *Machining Dynamics*, Springer US, Boston, MA.

[2] Altintas, Y., 2012, *Manufacturing Automation Metal Cutting Vibrations and CNC Design*.

[3] Budak, E., and Altintas¸, Y., 1998, "Analytical Prediction of Chatter Stability in Milling—Part I: General Formulation," J Dyn Syst Meas Control, 120(1), pp. 22–30.

[4] Altintaş, Y., and Budak, E., 1995, "Analytical Prediction of Stability Lobes in Milling," CIRP Annals, 44(1), pp. 357–362.

[5] Altintas, Y., Stepan, G., Merdol, D., and Dombovari, Z., 2008, "Chatter Stability of Milling in Frequency and Discrete Time Domain," CIRP J Manuf Sci Technol, 1(1), pp. 35–44.

[6] Budak, E., and Altintas¸, Y., 1998, "Analytical Prediction of Chatter Stability in Milling—Part II: Application of the General Formulation to Common Milling Systems," J Dyn Syst Meas Control, 120(1), pp. 31–36.

[7] Insperger, T., and Stépán, G., 2000, "Stability of the Milling Process," Periodica Polytechnica Mechanical Engineering, 44(1), pp. 47–57.

[8] Insperger, T., Mann, B. P., Stépán, G., and Bayly, P. V., 2003, "Stability of Up-Milling and down-Milling, Part 1: Alternative Analytical Methods," Int J Mach Tools Manuf, 43(1), pp. 25–34.

[9] Insperger, T., Stépán, G., Bayly, P. V., and Mann, B. P., 2003, "Multiple Chatter Frequencies in Milling Processes," J Sound Vib, 262(2), pp. 333–345.

[10] Govekar, E., Gradišek, J., Kalveram, M., Insperger, T., Weinert, K., Stépàn, G., and Grabec, I., 2005, "On Stability and Dynamics of Milling at Small Radial Immersion," CIRP Annals, 54(1), pp. 357–362.

[11] Insperger, T., and Stépán, G., 2004, "Updated Semi-Discretization Method for Periodic Delay-Differential Equations with Discrete Delay," Int J Numer Methods Eng, 61(1), pp. 117–141.

[12] Karandikar, J., Saleeby, K., Feldhausen, T., Kurfess, T., Schmitz, T., and Smith, S., 2022, "Evaluation of Automated Stability Testing in Machining through Closed-Loop Control and Bayesian Machine Learning," Mech Syst Signal Process, 181, p. 109531.

[13] Insperger, T., Mann, B. P., Stépán, G., and Bayly, P. V., 2003, "Stability of Up-Milling and down-Milling, Part 1: Alternative Analytical Methods," Int J Mach Tools Manuf, 43(1), pp. 25–34.

[14] Olvera, D., Elías-Zúñiga, A., Martínez-Alfaro, H., López de Lacalle, L. N., Rodríguez, C. A., and Campa, F. J., 2014, "Determination of the Stability Lobes in Milling Operations Based on Homotopy and Simulated Annealing Techniques," Mechatronics, 24(3), pp. 177–185.

[15] Postel, M., Bugdayci, B., and Wegener, K., 2020, "Ensemble Transfer Learning for Refining Stability Predictions in Milling Using Experimental Stability States," The International Journal of Advanced Manufacturing Technology, 107(9–10), pp. 4123–4139.

[16] Budak, E., Altintas¸, Y., and Armarego, E. J. A., 1996, "Prediction of Milling Force Coefficients From Orthogonal Cutting Data," J Manuf Sci Eng, 118(2), pp. 216–224.

[17] Delio, T., Tlusty, J., and Smith, S., 1992, "Use of Audio Signals for Chatter Detection and Control," Journal of Engineering for Industry, 114(2), pp. 146–157.

[18] Caliskan, H., Kilic, Z. M., and Altintas, Y., 2018, “On-Line Energy-Based Milling Chatter Detection,” Journal of Manufacturing Science and Engineering, Transactions of the ASME, 140(11), pp. 1–12.
[19] Rahimi, M. H., Huynh, H. N., and Altintas, Y., 2021, “On-Line Chatter Detection in Milling with Hybrid Machine Learning and Physics-Based Model,” CIRP J Manuf Sci Technol, 35, pp. 25–40.
[20] Chen, Y., Li, H., Hou, L., and Bu, X., 2019, “Feature Extraction Using Dominant Frequency Bands and Time-Frequency Image Analysis for Chatter Detection in Milling,” Precis Eng, 56, pp. 235–245.
[21] Zaeh, M. F., Schnoes, F., Obst, B., and Hartmann, D., 2020, “Combined Offline Simulation and Online Adaptation Approach for the Accuracy Improvement of Milling Robots,” CIRP Annals, 69(1), pp. 337–340.
[22] van Dijk, N. J. M., Doppenberg, E. J. J., Faassen, R. P. H., van de Wouw, N., Oosterling, J. A. J., and Nijmeijer, H., 2010, “Automatic In-Process Chatter Avoidance in the High-Speed Milling Process,” J Dyn Syst Meas Control, 132(3).
[23] Al-Regib, E., and Ni, J., 2010, “Chatter Detection in Machining Using Nonlinear Energy Operator,” J Dyn Syst Meas Control, 132(3).
[24] Kakinuma, Y., Sudo, Y., and Aoyama, T., 2011, “Detection of Chatter Vibration in End Milling Applying Disturbance Observer,” CIRP Annals, 60(1), pp. 109–112.
[25] Schmitz, T. L., 2003, “Chatter Recognition by a Statistical Evaluation of the Synchronously Sampled Audio Signal,” J Sound Vib, 262(3), pp. 721–730.
[26] Insperger, T., Mann, B. P., Surmann, T., and Stépán, G., 2008, “On the Chatter Frequencies of Milling Processes with Runout,” Int J Mach Tools Manuf, 48(10), pp. 1081–1089.
[27] Tarng, Y. S., and Li, T. C., 1994, “The Change of Spindle Speed for the Avoidance of Chatter in End Milling,” J Mater Process Technol, 41(2), pp. 227–236.
[28] Seguy, S., Insperger, T., Arnaud, L., Dessein, G., and Peigné, G., 2011, “SUPPRESSION OF PERIOD DOUBLING CHATTER IN HIGH-SPEED MILLING BY SPINDLE SPEED VARIATION,” Machining Science and Technology, 15(2), pp. 153–171.
[29] Shimoda, T., and Fujimoto, H., 2017, “Adaptive Spindle-Speed Selection for Chatter Avoidance to Achieve High-Precision NC Machining Based on Semi-Discretization Method,” *IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society*, IEEE, pp. 6709–6714.
[30] Maulimov, M., and Sencer, B., 2018, “Effect of Directional Relations on Milling Chatter Stability and Development of a Stability Index,” Procedia Manuf, 26, pp. 372–382.
[31] Munoa, J., Beudaert, X., Dombovari, Z., Altintas, Y., Budak, E., Brecher, C., and Stepan, G., 2016, “Chatter Suppression Techniques in Metal Cutting,” CIRP Annals, 65(2), pp. 785–808.
[32] Altintaş, Y., and Budak, E., 1995, “Analytical Prediction of Stability Lobes in Milling,” CIRP Annals, 44(1), pp. 357–362.
[33] Honeycutt, A., and Schmitz, T., 2017, “A Numerical and Experimental Investigation of Period-n Bifurcations in Milling,” J Manuf Sci Eng, 139(1).
[34] Bahtiyar, K., and Sencer, B., 2022, “Lightweight and Robust Chatter Detection Algorithms for Milling,” Manuf Lett, 33, pp. 388–394.
[35] Kuljanic, E., Sortino, M., and Totis, G., 2008, “Multisensor Approaches for Chatter Detection in Milling,” J Sound Vib, 312(4–5), pp. 672–693.

[36] Tran, M. Q., and Liu, M. K., 2019, "Chatter Identification in End Milling Process Based on Cutting Force Signal Processing," IOP Conf Ser Mater Sci Eng, 654(1), p. 012001.
[37] Liu, M.-K., Tran, Q. M., Qui, Y.-W., and Chung, C.-H., 2017, "Chatter Detection in Milling Process Based on Time-Frequency Analysis," *Volume 1: Processes*, American Society of Mechanical Engineers.
[38] Slavič, J., Mršnik, M., Česnik, M., Javh, J., and Boltežar, M., 2021, "Signal Processing," *Vibration Fatigue by Spectral Methods*, Elsevier, pp. 51–74.
[39] Koike Ryo, 2016, *THESIS - Realtime Monitoring and Stability Diagnosis of Cutting Process by Applying Disturbance Observer*.
[40] Chen, D., Zhang, X., Zhao, H., and Ding, H., 2021, "Development of a Novel Online Chatter Monitoring System for Flexible Milling Process," Mech Syst Signal Process, 159.
[41] Sun, Y., Liu, C., Sun, L., Xiong, Z., and Zhu, X., 2021, "Chatter Detection With Beat Effect Based on Beat Frequency Estimation," IEEE Transactions on Automation Science and Engineering, 18(4), pp. 2166–2174.
[42] Wasserman, L., 2004, *All of Statistics*, Springer New York, New York, NY.
[43] Hero, A. O., 2008, "Statistical Methods for Signal Processing," Matrix, 2.
[44] Alexander, W., and Williams, C., 2017, "Fundamental DSP Concepts," *Digital Signal Processing*, Elsevier, pp. 19–157.
[45] Syms, C., 2019, "Principal Components Analysis," *Encyclopedia of Ecology*, Elsevier, pp. 566–573.
[46] Müller, W., Nocke, T., and Schumann, H., 2006, "Enhancing the Visualization Process with Principal Component Analysis to Support the Exploration of Trends," *Proceedings of the 2006 Asia-Pacific Symposium on Information Visualisation - Volume 60*, Australian Computer Society, Inc., AUS, pp. 121–130.
[47] Li, Z., Li, W., Zhao, X., and Zheng, X., 2018, "Feature Frequency Extraction Algorithm Based on Principal Component Analysis and Its Application," Zhendong Ceshi Yu Zhenduan/Journal of Vibration, Measurement and Diagnosis, 38(4), pp. 834–842.
[48] Guo, M., Li, W., Yang, Q., Zhao, X., and Tang, Y., 2020, "Amplitude Filtering Characteristics of Singular Value Decomposition and Its Application to Fault Diagnosis of Rotating Machinery," Measurement, 154, p. 107444.
[49] Budak, E., and Tunc, L. T., 2010, "Identification and Modeling of Process Damping in Turning and Milling Using a New Approach," CIRP Annals, 59(1), pp. 403–408.
[50] Bachrathy, D., Stépán, G., and Turi, J., 2011, "State Dependent Regenerative Effect in Milling Processes," J Comput Nonlinear Dyn, 6(4).
[51] Stepan, G., Szalai, R., Mann, B. P., Bayly, P. v., Insperger, T., Gradisek, J., and Govekar, E., 2005, "Nonlinear Dynamics of High-Speed Milling—Analyses, Numerics, and Experiments," J Vib Acoust, 127(2), pp. 197–203.
[52] Kline, W. A., and DeVor, R. E., 1983, "The Effect of Runout on Cutting Geometry and Forces in End Milling," International Journal of Machine Tool Design and Research, 23(2–3), pp. 123–140.
[53] Seguy, S., Insperger, T., Arnaud, L., Dessein, G., and Peigné, G., 2010, "On the Stability of High-Speed Milling with Spindle Speed Variation," The International Journal of Advanced Manufacturing Technology, 48(9–12), pp. 883–895.
[54] Hunnekens, B., Dino, A. di, van de Wouw, N., van Dijk, N., and Nijmeijer, H., 2015, "Extremum-Seeking Control for the Adaptive Design of Variable Gain

Controllers," IEEE Transactions on Control Systems Technology, 23(3), pp. 1041–1051.

[55] Ariyur, K. B., and Krstić, M., 2003, *Real-Time Optimization by Extremum-Seeking Control,* John Wiley & Sons, Inc., Hoboken, NJ, USA.

[56] Hunnekens, B. G. B., Haring, M. A. M., van de Wouw, N., and Nijmeijer, H., 2014, "A Dither-Free Extremum-Seeking Control Approach Using 1st-Order Least-Squares Fits for Gradient Estimation," *53rd IEEE Conference on Decision and Control*, IEEE, pp. 2679–2684.