Journal of Sound and Vibration 571 (2024) 117954

Contents lists available at ScienceDirect

Journal of Sound and Vibration

journal homepage: www.elsevier.com/locate/jsvi

Stability prediction via parameter estimation from milling time series

James D. Turner *, Samuel A. Moore, Brian P. Mann

Department of Mechanical Engineering & Materials Science, Duke University, Durham, NC, 27708, USA

ARTICLE INFO

Keywords:
Milling
Stability
Parameter estimation
Time series
Delay differential equations

ABSTRACT

Machine tool vibrations impose severe limitations on industry. Recent progress in solving for the stability behavior of delay differential equations and in modeling milling operations with time delay differential equations has provided the potential to significantly reduce the aforementioned limitations. However, industry has yet to widely adopt the current academic knowledge due to the cost barriers in implementing this knowledge. Some of these cost prohibitive tasks include time-consuming experimental cutting tests used to calibrate model force parameters and experimental modal tests for every combination of tool, tool holder, tool length, spindle, and machine. This paper introduces an alternative approach whereby the vibration behavior of a milling tool during cutting is used to obtain the necessary model parameters for the common delay differential equation models of milling.

1. Introduction

Milling is one of the most commonly used material removal processes. In this process, a rotating cutting tool is gradually fed into a workpiece and the cutting teeth remove small chips of material with the goal of producing a smooth surface at a desired location. However, vibrations of the cutting tool can cause variations in the chip thickness, a poor surface finish, and reduced accuracy in the machined surface. Due to the relative motion between the cutting tool and workpiece, each tooth passage affects the cutting forces and chip thickness during the subsequent tooth passage. For certain combinations of cutting parameters, tool vibrations can cause a self-excited type of instability known as chatter. The large forces in chatter are problematic as they can damage the tool and produce a poorly machined, inaccurate surface. Thus, it is important to avoid chatter vibrations and to be able predict the machine settings that will give rise to this undesirable phenomenon.

Several past works have developed mathematical models for milling along with solution approaches to obtain stability charts from these delay differential equation models, e.g. [1–9]. However, all of these models require parameter values, terms describing the structural dynamics and cutting forces, that can drastically vary for even small differences in the cutting tool, tool holder, spindle, machine, workpiece material, etc. [10–12]. Measuring the parameters directly requires time-consuming tests with a skilled engineer and specialized equipment [11–14]. In contrast to the traditional approach of time-consuming modal tests and separate cutting force measurements to estimate model parameters, it would far better if the vibration time series from actual cutting tests could be used to estimate model parameters. This has the potential to enable the milling machine to quickly collect data and then run an algorithm to generate stability charts for chatter-free machining. This article takes aim at this problem and one approach to accomplish this. The proposed methodology is based on the idea of trajectory matching but is specifically adapted to the problem of milling.

* Corresponding author.
E-mail addresses: jdt40@turner.link (J.D. Turner), sam.a.moore@duke.edu (S.A. Moore), brian.mann@duke.edu (B.P. Mann).

https://doi.org/10.1016/j.jsv.2023.117954
Received 3 January 2023; Received in revised form 30 June 2023; Accepted 17 July 2023
Available online 20 July 2023
0022-460X/Published by Elsevier Ltd.

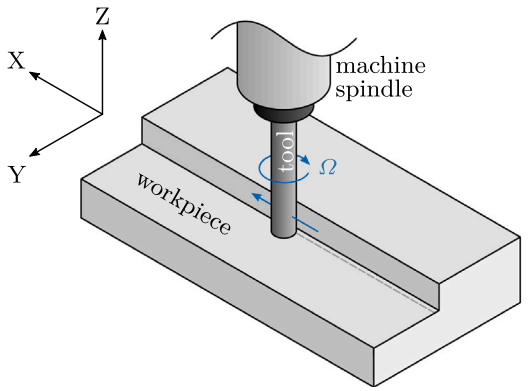


Fig. 1. Schematic of down-milling with an end mill. The directions of the rotation and feed of the tool are indicated with blue arrows. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Early work on the vibration of machine tools provided models to explain chatter, including the development of stability charts which compactly represent stability as a function of the system parameters, such as the spindle speed and cutting depth, and identifying regions of stability where material could be removed more quickly by cutting at higher spindle speeds [15–17]. Related works expanding on those efforts have investigated refinements in the models, experimental validation, and improvements in the analysis techniques [2,15–34]. The physics-based models which describe the dynamics of machining are typically written as delay differential equations (DDEs) [1,3,7,35–41]. Unlike ordinary differential equations (ODEs), even simple systems with time delay are infinite-dimensional, making analysis non-trivial [42]. Moreover, milling is an interrupted phenomenon, resulting in non-smooth and non-autonomous DDE models. Researchers have investigated methods to analytically or numerically approximate the solutions of these DDEs, primarily with a focus on stability [2–5,8,43–47]. Recent works have also investigated methods for optimization of machining parameters, e.g. [48,49].

The undesirable effects of chatter have also resulted in other creative attempts to avoid chatter vibrations. For example, researchers have developed active and passive vibration suppression methods to thwart chatter including spindle speed variation [50, 51], tuned mass dampers [52–54], and parametric excitation [55], among others. While useful, chatter suppression methods typically depend upon dynamic models with parameters that are subject to change or difficult to obtain. In addition, a formidable number of investigations have centered on chatter detection [56–62]. However, chatter detection methods provide limited system information. Thus, selecting machine settings, like spindle speed and cutting depth, from chatter detection methods results in a trial and error process.

Despite the fact that researchers have developed detailed models and sophisticated analysis techniques to address vibrations in machining, the industrial implementation of this knowledge has been limited. The primary hurdle for widespread adoption by industry is the costly and time-consuming traditional methods for estimating the model parameters that describe the structural dynamics and cutting forces; obtaining these parameters is time-consuming, difficult, and often requires the expertise of a skilled engineer. This has motivated several recent works that take aim at estimating machining model parameters from real cutting conditions [40,63–65]. The present work was inspired by Ref. [40], which sought to estimate parameters for a one-degree-of-freedom milling model from vibration time series collected during a milling process. This article expands on that work by estimating parameters for a two-degree-of-freedom milling model based on experimental data taken from Refs. [2,3]; it presents a new trajectory-matching approach specifically designed for time series collected from an instrumented milling process.

For the first time, this article presents a method to estimate the parameters for a two-degree-of-freedom DDE milling model directly from time series collected on an instrumented milling machine. It combines and extends models from [2,3] into an updated model which incorporates steady-state vibration of the tool outside of cutting, due to effects such as mass imbalance or misalignment of the cutting tool in the tool holder. Appendix B adapts and extends the spectral element method [66] to efficiently predict the trajectory of the tool using this model. Section 4 describes a method to estimate the model parameters from time series data. Finally, this article validates the proposed methods using experimental data and evaluates the quality of the stability predictions.

2. Dynamics model

Fig. 1 provides a schematic of the down-milling process. As the cutting tool rotates and is fed into the workpiece, it removes material from the workpiece. The tool also bends and vibrates due to the cutting forces. Considering only the first mode of vibration of the tool in the X and Y directions of Fig. 1, the 2-D vibration of the tool tip in the X and Y directions can be modeled by [2]:

$$\boldsymbol{M}\ddot{\boldsymbol{q}}(t) + \boldsymbol{C}\dot{\boldsymbol{q}}(t) + \boldsymbol{K}\boldsymbol{q}(t) = \boldsymbol{f}_{\mathrm{c}}(t) \tag{1}$$

where the two components of $\boldsymbol{q}(t) = \begin{bmatrix} q_{\mathrm{X}}(t) & q_{\mathrm{Y}}(t) \end{bmatrix}^{\mathrm{T}}$ are the modal displacements of the tool tip in the X and Y directions, respectively, relative to the axis of rotation, at time t; $\boldsymbol{M}$, $\boldsymbol{C}$, and $\boldsymbol{K}$ are mass, damping, and stiffness matrices of the tool; and $\boldsymbol{f}_{\mathrm{c}}(t)$ is the vector of cutting forces in the X and Y directions exerted by the workpiece on the tool near the tool tip at time t.

The cutting forces can be modeled by [2,3]:

$$\boldsymbol{f}_{\mathrm{c}}(t) = \boldsymbol{K}_{\mathrm{c}}(t)\left(\boldsymbol{q}(t) - \boldsymbol{q}(t-\tau)\right) + \boldsymbol{f}_0(t) \tag{2}$$

where $\boldsymbol{K}_{\mathrm{c}}(t)$ is a time-periodic coefficient for cutting forces which depend on the variation in the chip thickness; $\boldsymbol{f}_0(t)$ is a time-periodic vector of additional cutting forces; and τ is the tooth passing period. The value of τ is given by $\tau = 2\pi\Omega^{-1}n_{\mathrm{teeth}}^{-1}$, where Ω is the spindle speed in rad/s, and n_{teeth} is the number of teeth on the tool. Section 2.1 discusses the cutting forces in more detail.

Eq. (1) assumes that the steady-state position of the tool in the absence of cutting forces is perfectly aligned to and centered about the axis of rotation. However, this assumption may not be valid. For example, the rotating components, such as the tool and tool holder, may have a mass imbalance which can cause vibration of the tool; the tool could also be imperfectly positioned in the tool holder. These effects can be approximated by an additional, time-periodic, term $\boldsymbol{f}_{\mathrm{u}}(t)$:

$$\boldsymbol{M}\ddot{\boldsymbol{q}}(t) + \boldsymbol{C}\boldsymbol{q}(t) + \boldsymbol{K}\boldsymbol{q}(t) = \boldsymbol{f}_{\mathrm{c}}(t) + \boldsymbol{f}_{\mathrm{u}}(t) \tag{3}$$

Substituting Eq. (2) and rearranging the terms, Eq. (3) can be written as a first-order DDE:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(t,\boldsymbol{p})\boldsymbol{x}(t) + \boldsymbol{B}(t,\boldsymbol{p})\boldsymbol{x}(t-\tau) + \boldsymbol{c}(t,\boldsymbol{p}) \tag{4}$$

where

$$\boldsymbol{x}(t) = \begin{bmatrix} \boldsymbol{q}(t) \\ \dot{\boldsymbol{q}}(t) \end{bmatrix} \tag{5}$$

$$\boldsymbol{A}(t,\boldsymbol{p}) = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} \\ -\boldsymbol{M}^{-1}\boldsymbol{K} + \boldsymbol{M}^{-1}\boldsymbol{K}_{\mathrm{c}}(t) & -\boldsymbol{M}^{-1}\boldsymbol{C} \end{bmatrix} \tag{6}$$

$$\boldsymbol{B}(t,\boldsymbol{p}) = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ -\boldsymbol{M}^{-1}\boldsymbol{K}_{\mathrm{c}}(t) & \boldsymbol{0} \end{bmatrix} \tag{7}$$

$$\boldsymbol{c}(t,\boldsymbol{p}) = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{M}^{-1}\boldsymbol{f}_0(t) + \boldsymbol{M}^{-1}\boldsymbol{f}_{\mathrm{u}}(t) \end{bmatrix} \tag{8}$$

The $\boldsymbol{f}_{\mathrm{u}}(t)$ term is modeled by:

$$\boldsymbol{f}_{\mathrm{u}}(t) = \boldsymbol{f}_{\mathrm{u,0}} + \Re\left[\boldsymbol{f}_{\mathrm{u,p}}\mathrm{e}^{\mathrm{i}\Omega t}\right] \tag{9}$$

where $\boldsymbol{f}_{\mathrm{u,0}}$ is a constant, real vector, where the subscript '0' refers to this being the constant term; $\boldsymbol{f}_{\mathrm{u,p}}$ is a constant, complex vector, where the subscript 'p' refers to this being the coefficient of the periodic term; and $\Re : \mathbb{C} \to \mathbb{R}$ extracts the real part of its argument. Note that $\boldsymbol{f}_{\mathrm{u}}(t)$ affects the trajectory of the system, so it is important for fitting time series, but it does not affect the characteristic multipliers (CMs) or stability. This can be seen most easily using the spectral element method; see Section 2.3 and Appendix B.

2.1. Cutting forces

The derivation presented here of the cutting forces for a helical cutting tool is similar to [3], but this derivation provides simplified expressions for $\boldsymbol{K}_{\mathrm{c}}(t)$ and $\boldsymbol{f}_0(t)$, and it incorporates edge coefficients as in [2]. The cutting forces are distributed along a small contact region, as shown in Fig. 2(a). The total cutting forces on the tool in the X and Y directions can be computed by summing over the teeth and integrating over the contact region for each tooth:

$$\boldsymbol{f}_{\mathrm{c}}(t) = \sum_{k=1}^{n_{\mathrm{teeth}}} g_k(t) \int_{z_{k,\mathrm{lo}}(t)}^{z_{k,\mathrm{hi}}(t)} \left(\begin{bmatrix} -\cos\theta_k(z,t) \\ \sin\theta_k(z,t) \end{bmatrix} f_{\mathrm{t}}(z,t) + \begin{bmatrix} -\sin\theta_k(z,t) \\ -\cos\theta_k(z,t) \end{bmatrix} f_{\mathrm{n}}(z,t) \right) \mathrm{d}z \tag{10}$$

where $f_{\mathrm{n}}(z,t)$ and $f_{\mathrm{t}}(z,t)$ are the normal and tangential cutting forces per unit height at height z and time t, as illustrated in Fig. 2(b); $z_{k,\mathrm{lo}}(t)$ is the lowest position where tooth k is in contact with the workpiece; $z_{k,\mathrm{hi}}(t)$ is the highest position where tooth k is in contact with the workpiece; $g_k(t)$ is 1 if tooth k is cutting, and 0 otherwise; and the instantaneous angle of the point at height z on tooth k at time t is given by

$$\theta_k(z,t) = \theta_k(z,t_0) + \Omega \cdot (t - t_0) \tag{11}$$

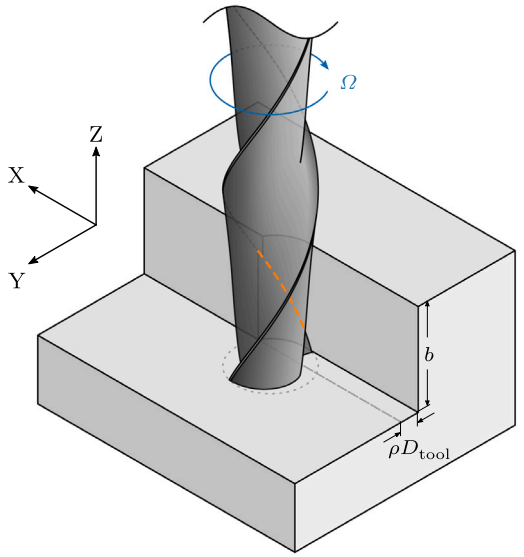
For points in the contact region, the radial chip thickness can be approximated by

$$w_k(z,t) = h\sin\theta_k(z,t) + \begin{bmatrix} \sin\theta_k(z,t) & \cos\theta_k(z,t) \end{bmatrix}\left(\boldsymbol{q}(t) - \boldsymbol{q}(t-\tau)\right) \tag{12}$$

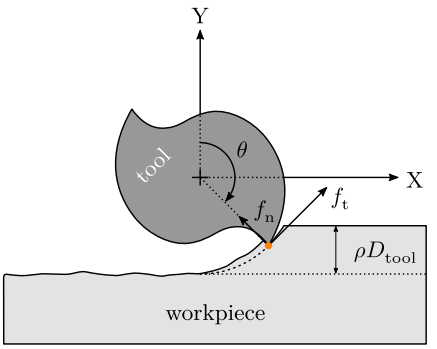
where h is the feed per tooth. The cutting forces per unit height can be modeled by

$$f_{\mathrm{t}}(z,t) = K_{\mathrm{t}}w_k(z,t) + K_{\mathrm{te}} \tag{13}$$

$$f_{\mathrm{n}}(z,t) = K_{\mathrm{n}}w_k(z,t) + K_{\mathrm{ne}} \tag{14}$$



(a) Isometric view of the tool and workpiece. The contact region between the tooth and the workpiece is indicated with a dashed orange line.



(b) Cross section of the tool and workpiece, viewed from above. The arrows labeled f_t and f_n show the tangential and normal cutting forces per unit height at this height and instant in time.

Fig. 2. Illustrations of the contact region and the cutting forces per unit height for a helical cutting tool with two teeth.

where K_t and K_n are the cutting pressures, and K_{te} and K_{ne} are the edge coefficients. Substituting and rearranging, Eq. (10) becomes Eq. (2), with

$$\boldsymbol{K}_c(t) = \sum_{k=1}^{n_{teeth}} g_k(t) \int_{z_{k,lo}(t)}^{z_{k,hi}(t)} \begin{bmatrix} -K_t sc - K_n s^2 & -K_t c^2 - K_n sc \\ K_t s^2 - K_n sc & K_t sc - K_n c^2 \end{bmatrix} \mathrm{d}z \tag{15}$$

$$\boldsymbol{f}_0(t) = \sum_{k=1}^{n_{teeth}} g_k(t) \int_{z_{k,lo}(t)}^{z_{k,hi}(t)} \left(h \begin{bmatrix} -K_t sc - K_n s^2 \\ K_t s^2 - K_n sc \end{bmatrix} + \begin{bmatrix} -K_{te} c - K_{ne} s \\ K_{te} s - K_{ne} c \end{bmatrix} \right) \mathrm{d}z \tag{16}$$

where

$$s = \sin\theta_k(z,t) \qquad c = \cos\theta_k(z,t) \tag{17}$$

To evaluate Eqs. (15) and (16), it is necessary to compute $g_k(t)$ and the bounds of integration. For down-milling, the leading tip of a tooth enters the cut at angle $\theta_{en} = \pi - \arccos(1-2\rho)$ and exits the cut at angle $\theta_{ex} = \pi$, where the radial immersion is ρD_{tool}, and D_{tool} is the diameter of the tool. Note, however, that for a tool with a helix angle $\beta \neq 0$, the upper contact point of the tooth does not start cutting until a short time after the tip starts cutting, and it does not exit the cut until a short time after the tip of the tooth exits the cut. Let

$$\kappa = \frac{\tan\beta}{D_{tool}/2} \tag{18}$$

so that

$$\theta_k(z,t) = \theta_k(0,t) - \kappa z \tag{19}$$

Then, for axial cut depth b, the tool rotates through an angle of $\theta_{lag} = \kappa b$ between the tip of a tooth entering the cut and the upper contact point on the tooth entering the cut, and between the tip of a tooth exiting the cut and the upper contact point on the tooth exiting the cut. This is illustrated in Fig. 3. So, the contact indicator function for tooth k is

$$g_k(t) = \begin{cases} 1 & \text{if } \theta_k(0,t) \in [\theta_{en}, \theta_{ex} + \theta_{lag}] \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

The heights of the instantaneous lower and upper points of contact between tooth k and the workpiece are given by

$$z_{k,\mathrm{lo}}(t) = \begin{cases} 0 & \text{if } \theta_k(0,t) \in \left[\theta_{\mathrm{en}}, \theta_{\mathrm{ex}}\right] \\ -\kappa^{-1}\left(\theta_{\mathrm{ex}} - \theta_k(0,t)\right) & \text{if } \theta_k(0,t) \in \left[\theta_{\mathrm{ex}}, \theta_{\mathrm{ex}} + \theta_{\mathrm{lag}}\right] \end{cases} \tag{21}$$

$$z_{k,\mathrm{hi}}(t) = \begin{cases} -\kappa^{-1}\left(\theta_{\mathrm{en}} - \theta_k(0,t)\right) & \text{if } \theta_k(0,t) \in \left[\theta_{\mathrm{en}}, \theta_{\mathrm{en}} + \theta_{\mathrm{lag}}\right] \\ b & \text{if } \theta_k(0,t) \in \left[\theta_{\mathrm{en}} + \theta_{\mathrm{lag}}, \theta_{\mathrm{ex}} + \theta_{\mathrm{lag}}\right] \end{cases} \tag{22}$$

and the corresponding angles for the lower and upper points of contact are

$$\theta_{k,\mathrm{lo}}(t) = \theta_k\left(z_{k,\mathrm{lo}}(t), t\right) = \begin{cases} \theta_k(0,t) & \text{if } \theta_k(0,t) \in \left[\theta_{\mathrm{en}}, \theta_{\mathrm{ex}}\right] \\ \theta_{\mathrm{ex}} & \text{if } \theta_k(0,t) \in \left[\theta_{\mathrm{ex}}, \theta_{\mathrm{ex}} + \theta_{\mathrm{lag}}\right] \end{cases} \tag{23}$$

$$\theta_{k,\mathrm{hi}}(t) = \theta_k\left(z_{k,\mathrm{hi}}(t), t\right) = \begin{cases} \theta_{\mathrm{en}} & \text{if } \theta_k(0,t) \in \left[\theta_{\mathrm{en}}, \theta_{\mathrm{en}} + \theta_{\mathrm{lag}}\right] \\ \theta_k(0,t) - \theta_{\mathrm{lag}} & \text{if } \theta_k(0,t) \in \left[\theta_{\mathrm{en}} + \theta_{\mathrm{lag}}, \theta_{\mathrm{ex}} + \theta_{\mathrm{lag}}\right] \end{cases} \tag{24}$$

Then, Eqs. (15) and (16) can be simplified to

$$\boldsymbol{K}_{\mathrm{c}}(t) = \sum_{k=1}^{n_{\mathrm{teeth}}} g_k(t) \begin{bmatrix} -K_{\mathrm{t}}\overline{sc} - K_{\mathrm{n}}\overline{s^2} & -K_{\mathrm{t}}\overline{c^2} - K_{\mathrm{n}}\overline{sc} \\ K_{\mathrm{t}}\overline{s^2} - K_{\mathrm{n}}\overline{sc} & K_{\mathrm{t}}\overline{sc} - K_{\mathrm{n}}\overline{c^2} \end{bmatrix} \tag{25}$$

$$\boldsymbol{f}_0(t) = \sum_{k=1}^{n_{\mathrm{teeth}}} g_k(t) \left(h \begin{bmatrix} -K_{\mathrm{t}}\overline{sc} - K_{\mathrm{n}}\overline{s^2} \\ K_{\mathrm{t}}\overline{s^2} - K_{\mathrm{n}}\overline{sc} \end{bmatrix} + \begin{bmatrix} -K_{\mathrm{te}}\overline{c} - K_{\mathrm{ne}}\overline{s} \\ K_{\mathrm{te}}\overline{s} - K_{\mathrm{ne}}\overline{c} \end{bmatrix} \right) \tag{26}$$

where, for t such that $g_k(t) = 1$,

$$\overline{s} = \int_{z_0(t)}^{z_1(t)} \sin\theta_k(z,t)\,\mathrm{d}z = \begin{cases} b\sin\theta_k(0,t) & \text{if } \kappa = 0 \\ -\frac{1}{\kappa}\left(-\cos\theta\right)\Big|_{\theta=\theta_{k,\mathrm{lo}}(t)}^{\theta=\theta_{k,\mathrm{hi}}(t)} & \text{if } \kappa \neq 0 \end{cases} \tag{27}$$

$$\overline{c} = \int_{z_0(t)}^{z_1(t)} \cos\theta_k(z,t)\,\mathrm{d}z = \begin{cases} b\cos\theta_k(0,t) & \text{if } \kappa = 0 \\ -\frac{1}{\kappa}\sin\theta\Big|_{\theta=\theta_{k,\mathrm{lo}}(t)}^{\theta=\theta_{k,\mathrm{hi}}(t)} & \text{if } \kappa \neq 0 \end{cases} \tag{28}$$

$$\overline{s^2} = \int_{z_0(t)}^{z_1(t)} \sin^2\theta_k(z,t)\,\mathrm{d}z = \begin{cases} b\sin^2\theta_k(0,t) & \text{if } \kappa = 0 \\ -\frac{1}{\kappa}\left(\frac{1}{2}\left(\theta - \sin\theta\cos\theta\right)\right)\Big|_{\theta=\theta_{k,\mathrm{lo}}(t)}^{\theta=\theta_{k,\mathrm{hi}}(t)} & \text{if } \kappa \neq 0 \end{cases} \tag{29}$$

$$\overline{c^2} = \int_{z_0(t)}^{z_1(t)} \cos^2\theta_k(z,t)\,\mathrm{d}z = \begin{cases} b\cos^2\theta_k(0,t) & \text{if } \kappa = 0 \\ -\frac{1}{\kappa}\left(\frac{1}{2}\left(\theta + \sin\theta\cos\theta\right)\right)\Big|_{\theta=\theta_{k,\mathrm{lo}}(t)}^{\theta=\theta_{k,\mathrm{hi}}(t)} & \text{if } \kappa \neq 0 \end{cases} \tag{30}$$

$$\overline{sc} = \int_{z_0(t)}^{z_1(t)} \sin\theta_k(z,t)\cos\theta_k(z,t)\,\mathrm{d}z = \begin{cases} b\sin\theta_k(0,t)\cos\theta_k(0,t) & \text{if } \kappa = 0 \\ -\frac{1}{\kappa}\left(-\frac{1}{2}\cos^2\theta\right)\Big|_{\theta=\theta_{k,\mathrm{lo}}(t)}^{\theta=\theta_{k,\mathrm{hi}}(t)} & \text{if } \kappa \neq 0 \end{cases} \tag{31}$$

2.2. Analytical solution for vibration between cuts

When no teeth are in contact with the workpiece, Eq. (4) simplifies to an ODE, which can be solved analytically:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_{\mathrm{o}}(\boldsymbol{p})\boldsymbol{x}(t) + \boldsymbol{c}_{\mathrm{o}}(t,\boldsymbol{p}) \qquad \forall t \text{ s.t.} \left(g_k(t) = 0 \ \forall k \in \left\{1, \ldots, n_{\mathrm{teeth}}\right\}\right) \tag{32}$$

where

$$\boldsymbol{A}_{\mathrm{o}}(\boldsymbol{p}) = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} \\ -\boldsymbol{M}^{-1}\boldsymbol{K} & -\boldsymbol{M}^{-1}\boldsymbol{C} \end{bmatrix} \qquad \boldsymbol{c}_{\mathrm{o}}(t,\boldsymbol{p}) = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{M}^{-1}\boldsymbol{f}_{\mathrm{u}}(t) \end{bmatrix} \tag{33}$$

It can be shown that the solution to Eq. (32), using Eq. (9), is

$$\boldsymbol{x}(t) = \boldsymbol{x}_{\mathrm{ss}}(t) + \boldsymbol{x}_{\mathrm{h}}(t) \qquad \forall t \text{ s.t.} \left(g_k(t) = 0 \ \forall k \in \left\{1, \ldots, n_{\mathrm{teeth}}\right\}\right) \tag{34}$$

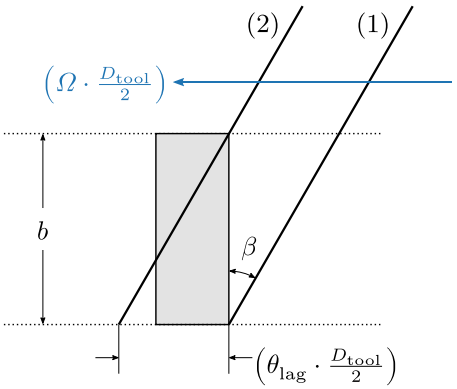


Fig. 3. Illustration of the helix and lag angles. The view is looking at the side of the tool from the point of view of the workpiece, with the helix unrolled onto a plane. The thick diagonal lines are the helical edge of the tooth (1) when the tip of the tooth starts cutting and (2) when the point at height b on the tooth starts cutting. The tool rotates through an angle of θ_{lag} between instants (1) and (2). The shaded rectangle indicates the size of the chip being cut on this tooth passage; the size varies with the axial cut depth b, which determines the height, and the radial immersion, which affects the width. At any instant during the cut, the contact between the tooth and the workpiece is the intersection between a diagonal line representing the tooth edge and the shaded rectangle.

where

$$\boldsymbol{x}_{\text{h}}(t) = \mathrm{e}^{\boldsymbol{A}_{\text{o}}(\boldsymbol{p})(t-t_0)}\left(\boldsymbol{x}(t_0) - \boldsymbol{x}_{\text{ss}}(t_0)\right) \tag{35}$$

$$\boldsymbol{x}_{\text{ss}}(t) = \boldsymbol{x}_{\text{ss,0}} + \Re\left[\boldsymbol{x}_{\text{ss,p}}\mathrm{e}^{\mathrm{i}\Omega t}\right] \qquad \boldsymbol{x}_{\text{ss,0}} = -\left(\boldsymbol{A}_{\text{o}}(\boldsymbol{p})\right)^{-1}\begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{M}^{-1}\boldsymbol{f}_{\text{u,0}}\end{bmatrix} \qquad \boldsymbol{x}_{\text{ss,p}} = \left(\mathrm{i}\Omega\boldsymbol{I} - \boldsymbol{A}_{\text{o}}(\boldsymbol{p})\right)^{-1}\begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{M}^{-1}\boldsymbol{f}_{\text{u,p}}\end{bmatrix} \tag{36}$$

Note that this implies that

$$\boldsymbol{M}^{-1}\boldsymbol{f}_{\text{u,0}} = \begin{bmatrix}\boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\left(-\left(\boldsymbol{A}_{\text{o}}(\boldsymbol{p})\right)\boldsymbol{x}_{\text{ss,0}}\right) \tag{37}$$

$$\boldsymbol{M}^{-1}\boldsymbol{f}_{\text{u,p}} = \begin{bmatrix}\boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\left(\mathrm{i}\Omega\boldsymbol{I} - \boldsymbol{A}_{\text{o}}(\boldsymbol{p})\right)\boldsymbol{x}_{\text{ss,p}} \tag{38}$$

which provides a convenient way to obtain $\boldsymbol{M}^{-1}\boldsymbol{f}_{\text{u}}(t)$ from the steady-state solution between cuts $\boldsymbol{x}_{\text{ss}}(t)$.

2.3. Approximate solution using the spectral element method

The full solution of the DDE model (Eq. (4)) can be approximated in a computationally efficient way by an extended version of the spectral element method introduced by Khasawneh and Mann [66]. Let $\boldsymbol{y}(t)$ be this approximation of the state $\boldsymbol{x}(t)$. Time is split into segments of length τ, where each segment begins at the start of a cut and ends at the start of the next cut. The time interval during each cut is split into multiple elements. Within each element, the solution is approximated by polynomial interpolation between the values of $\boldsymbol{y}(t)$ at discrete times, known as nodes. Between cuts, the solution is described by Eq. (34), where the initial condition is the node at the end of the preceding cut. This is illustrated in Fig. 4, where the start of the initial cut is t_{i}, and the duration of each cut is t_{d}.

The spectral element method computes an approximate mapping from the nodes in one segment to the nodes in the next, which can be written as:

$$\begin{bmatrix}\boldsymbol{y}\left(t_{\text{i}} + (k+1)\tau + \eta_0\right)\\ \boldsymbol{y}\left(t_{\text{i}} + (k+1)\tau + \eta_1\right)\\ \vdots\\ \boldsymbol{y}\left(t_{\text{i}} + (k+1)\tau + \eta_\ell\right)\end{bmatrix} = \boldsymbol{Q}_{k+1}(\boldsymbol{p})\begin{bmatrix}\boldsymbol{y}(t_{\text{i}} + k\tau + \eta_0)\\ \boldsymbol{y}(t_{\text{i}} + k\tau + \eta_1)\\ \vdots\\ \boldsymbol{y}(t_{\text{i}} + k\tau + \eta_\ell)\end{bmatrix} + \boldsymbol{r}_{k+1}(\boldsymbol{p}), \qquad 0 \le \eta_0 \le \cdots \le \eta_\ell \le t_{\text{d}}, \quad k = 0, 1, \ldots \tag{39}$$

where $\boldsymbol{Q}_{k+1}(\boldsymbol{p})$ and $\boldsymbol{r}_{k+1}(\boldsymbol{p})$ represent the mapping from segment k to segment $k+1$, and $\eta_0, \ldots, \eta_\ell$ are the times of the each segment's nodes relative to the start of the segment. Due to the periodicity of $\boldsymbol{A}(t, \boldsymbol{p})$, $\boldsymbol{B}(t, \boldsymbol{p})$, and $\boldsymbol{c}(t, \boldsymbol{p})$ in Eq. (4) with respect to the spindle rotation period, the mapping is also periodic:

$$\boldsymbol{Q}_{k+n_{\text{teeth}}}(\boldsymbol{p}) = \boldsymbol{Q}_k(\boldsymbol{p}) \qquad \boldsymbol{r}_{k+n_{\text{teeth}}}(\boldsymbol{p}) = \boldsymbol{r}_k(\boldsymbol{p}) \tag{40}$$

So, given the solution at the initial nodes, $\boldsymbol{y}(t_{\text{i}} + \eta_0), \ldots, \boldsymbol{y}(t_{\text{i}} + \eta_\ell)$, a continuous approximation of the solution for the first n segments can be obtained by the following steps:

(1) Compute $\boldsymbol{Q}_k(\boldsymbol{p})$ and $\boldsymbol{r}_k(\boldsymbol{p})$ for $k = 1, \ldots, n_{\text{teeth}}$.
(2) Apply Eq. (39) repeatedly to compute the solution at the nodes for each of the first n segments.
(3) For each segment:

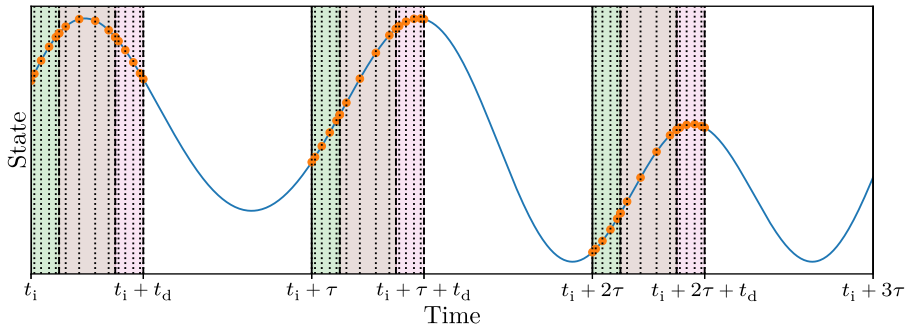


Fig. 4. Illustration of the spectral element segments, elements, and nodes for approximating the solution of Eq. (4). The state of the system is indicated with a curved blue line. The divisions between segments are indicated with solid vertical lines. The end of each element is indicated with a vertical dashed line. The location of each node is indicated with a vertical line (solid, dashed, or dotted), and the corresponding value of the state is indicated with an orange circular marker. The intervals of time during cuts have a shaded backgrounds (with a different color for each element within a segment), while the intervals of time between cuts have a white background. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

- Within the cut, use Lagrange polynomial interpolation between the nodes within each element.
- Following the cut, use the last node from the cut as the initial condition for Eq. (34).

The matrices $\boldsymbol{Q}_1(\boldsymbol{p}), \ldots, \boldsymbol{Q}_{n_{\text{teeth}}}(\boldsymbol{p})$ can also be used to compute the CMs, and thus stability, of the system.

For more details, see Appendix B. Eq. (4) corresponds to Eq. (B.1), and Eq. (34) corresponds to Eq. (B.2). It was found that, for this problem, a good choice for the number of elements and the polynomial order was $n_e = 3$ and $n_o = 5$. The boundaries of the elements were chosen to coincide with the times when the cutting forces were non-differentiable, i.e. when the tip of the tooth started and stopped cutting and when the point on the tooth at height b started and stopped cutting. This choice of element boundaries was beneficial because the spectral element approximation was infinitely differentiable within each element but only continuous across elements.

3. Experimental setup

The proposed model and methods were evaluated using data collected from test cuts on a 5-axis linear motor milling machine, cutting into an aluminum (7050-T7451) block, with various spindle speeds and cutting depths [2]. The experimental setup closely resembled the use of long slender tools by the aerospace industry to machine deep pocket monolithic structures. The tool was a carbide end mill with $n_{\text{teeth}} = 2$, diameter $D_{\text{tool}} = 12.75\,\text{mm}$, helix angle $\beta = 30°$, and 106 mm overhang. The radial immersion ratio was $\rho = 0.05$, and the feed rate was $h = 0.127\,\text{mm}$ per tooth. A clean-up pass to smooth the workpiece was performed between each test cut.

The milling machine was instrumented with sensors to collect time series data for each test cut. The time series data for each test cut consisted of a tachometer signal and noisy measurements of the position of the tool tip. Fig. 5 shows a photograph of the sensor setup. The position of the tool 19 mm from the tip was measured using two capacitive sensors held close to the tool in a rigid fixture; the position of the tool tip was estimated by scaling these measurements. The tachometer signal indicated at each instant in time whether a mark drawn on the tool was detected or not; this indirectly provided information about the spindle speed and orientation of the tool. The sampling frequency for the sensors was 25 kHz. For more details on the setup and data collection, see [2].

4. Fitting to time series data

The objective of this article is to estimate the unknown model parameters by fitting the model to the time series data, so that stability charts can be generated using the estimated parameters. In theory, it would be possible to estimate all of the unknowns simultaneously by trajectory matching, using a global optimization algorithm to fit the spectral element approximation of the milling model to the time series. However, this is difficult due to the large number of unknown parameters. Fortunately, the unknown parameters can instead be estimated in a sequence of stages, where each stage estimates a different subset of the parameters. For example, it is possible to estimate parameters describing $\boldsymbol{M}^{-1}\boldsymbol{K}$ and $\boldsymbol{M}^{-1}\boldsymbol{C}$ using the portions of the data between cuts, independently of the parameters describing the cutting forces. This approach of breaking up the overall problem into smaller subproblems limits the number of parameters which need to be estimated simultaneously, which significantly simplifies the optimization problems which need to be solved. This section describes how the model can be parameterized, provides a brief overview of the trajectory matching technique used in the last two stages, and then describes the stages of the parameter estimation strategy in sequence.

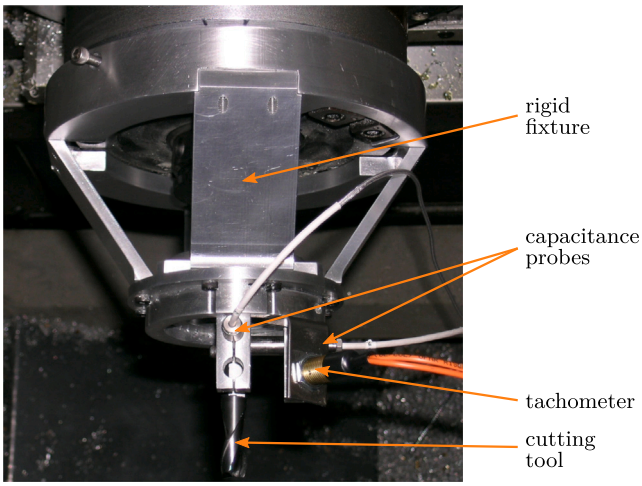


Fig. 5. Annotated photograph of the cutting tool and sensor setup. [Photograph by Brian P. Mann, 2003.]

4.1. Parameterizing the model

To make model fitting work well, it is beneficial to define the parameter vector $\boldsymbol{p}$ such that the number of unknown parameters to fit is minimized and the parameter values always meet physical constraints. This section describes the specific parameter choices, in the order which they were estimated. The details of how they were estimated are provided in the following sections.

First, the axial cut depth b, radial immersion ratio ρ, and feed h per tooth were assumed to be known but possibly different between time series; they correspond to settings on the milling machine:

$$p_1 = b \tag{41}$$

$$p_2 = \rho \tag{42}$$

$$p_3 = h \tag{43}$$

The number of teeth, diameter, and helix angle of the tool were assumed to be known and the same for all time series:

$$p_4 = n_{\text{teeth}} \tag{44}$$

$$p_5 = D_{\text{tool}} \tag{45}$$

$$p_6 = \beta \tag{46}$$

The spindle speed Ω should be approximately controlled by the milling machine, but, for higher accuracy, Ω was estimated from separate tachometer measurements. The time $t_{\text{up}}(0)$ of the first upward edge of the tachometer signal—which was used as an indicator of the phase of the tool—could also be estimated from the tachometer signal, as described in Section 4.3.

$$p_7 = \Omega \tag{47}$$

$$p_8 = t_{\text{up}}(0) \tag{48}$$

As described in Section 4.4, an initial estimate of the steady-state solution outside of cutting, $\boldsymbol{x}_{\text{ss}}(t)$ (Eq. (36)), was obtained by fitting the steady-state vibration data before cutting started. Let

$$\boldsymbol{q}_{\text{ss}}(t) = \boldsymbol{q}_{\text{ss},0} + \mathcal{R}\left[\boldsymbol{q}_{\text{ss,p}} e^{\mathrm{i}\Omega t}\right] \tag{49}$$

such that

$$\boldsymbol{x}_{\text{ss}}(t) = \begin{bmatrix} \boldsymbol{q}_{\text{ss}}(t) \\ \dot{\boldsymbol{q}}_{\text{ss}}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{q}_{\text{ss},0} \\ \boldsymbol{0} \end{bmatrix} + \mathcal{R}\left[\begin{bmatrix} \boldsymbol{q}_{\text{ss,p}} \\ \mathrm{i}\Omega \boldsymbol{q}_{\text{ss,p}} \end{bmatrix} e^{\mathrm{i}\Omega t}\right] \tag{50}$$

So, the coefficients of the steady-state solution in Eq. (36) are

$$\boldsymbol{x}_{\text{ss},0} = \begin{bmatrix} \boldsymbol{q}_{\text{ss},0} \\ \boldsymbol{0} \end{bmatrix} \qquad \boldsymbol{x}_{\text{ss,p}} = \begin{bmatrix} \boldsymbol{q}_{\text{ss,p}} \\ \mathrm{i}\Omega \boldsymbol{q}_{\text{ss,p}} \end{bmatrix} \tag{51}$$

Six real parameters are sufficient to describe the steady-state solution:

$$p_9 = \begin{bmatrix} 1 & 0 \end{bmatrix} \boldsymbol{q}_{\text{ss},0} \tag{52}$$

$$p_{10} = \begin{bmatrix} 0 & 1 \end{bmatrix} \boldsymbol{q}_{\mathrm{ss,0}} \tag{53}$$

$$p_{11} = \begin{bmatrix} 1 & 0 \end{bmatrix} \Re\left[\boldsymbol{q}_{\mathrm{ss,p}}\right] \qquad p_{12} = \begin{bmatrix} 1 & 0 \end{bmatrix} \Im\left[\boldsymbol{q}_{\mathrm{ss,p}}\right] \tag{54}$$

$$p_{13} = \begin{bmatrix} 0 & 1 \end{bmatrix} \Re\left[\boldsymbol{q}_{\mathrm{ss,p}}\right] \qquad p_{14} = \begin{bmatrix} 0 & 1 \end{bmatrix} \Im\left[\boldsymbol{q}_{\mathrm{ss,p}}\right] \tag{55}$$

Then,

$$\boldsymbol{q}_{\mathrm{ss,0}} = \begin{bmatrix} p_9 \\ p_{10} \end{bmatrix} \qquad \boldsymbol{q}_{\mathrm{ss,p}} = \begin{bmatrix} p_{11} + \mathrm{i}p_{12} \\ p_{13} + \mathrm{i}p_{14} \end{bmatrix} \tag{56}$$

Recall that, by Eqs. (9), (37), and (38), this also describes $\boldsymbol{M}^{-1}\boldsymbol{f}_{\mathrm{u}}(t)$. Note that these parameters differ between time series. In particular, the amplitude tends to increase with spindle speed, and the phase relative to $t = 0\,\mathrm{s}$ differs.

As described in Section 4.5, by fitting Eq. (32) to the vibration between cuts, it is possible to estimate the $\boldsymbol{M}^{-1}\boldsymbol{K}$ and $\boldsymbol{M}^{-1}\boldsymbol{C}$ matrices, refine the estimate of the steady-state solution $\boldsymbol{x}_{\mathrm{ss}}(t)$ (which also describes $\boldsymbol{M}^{-1}\boldsymbol{f}_{\mathrm{u}}(t)$), and estimate the phase of the cuts. These were parameterized as follows. First, it is reasonable to assume that the $\boldsymbol{M}$, $\boldsymbol{K}$, and $\boldsymbol{C}$ matrices are approximately diagonal, so $\boldsymbol{M}^{-1}\boldsymbol{K}$ and $\boldsymbol{M}^{-1}\boldsymbol{C}$ can be written as

$$\boldsymbol{M}^{-1}\boldsymbol{K} = \begin{bmatrix} \left(\boldsymbol{M}^{-1}\boldsymbol{K}\right)_{\mathrm{XX}} & 0 \\ 0 & \left(\boldsymbol{M}^{-1}\boldsymbol{K}\right)_{\mathrm{YY}} \end{bmatrix} \qquad \boldsymbol{M}^{-1}\boldsymbol{C} = \begin{bmatrix} \left(\boldsymbol{M}^{-1}\boldsymbol{C}\right)_{\mathrm{XX}} & 0 \\ 0 & \left(\boldsymbol{M}^{-1}\boldsymbol{C}\right)_{\mathrm{YY}} \end{bmatrix} \tag{57}$$

However, to be physically realistic, all the diagonal elements in these matrices should be positive, so it is beneficial to use the logarithm of a diagonal element as the corresponding parameter to ensure that the resulting diagonal element is positive. Additionally, the two diagonal elements in each matrix should be very similar to each other, so it can be beneficial for one parameter to describe the first element and the other parameter to describe the ratio between the two elements. In other words, the parameters used to describe $\boldsymbol{M}^{-1}\boldsymbol{K}$ and $\boldsymbol{M}^{-1}\boldsymbol{C}$ were the following:

$$p_{15} = \ln\left(\left(\boldsymbol{M}^{-1}\boldsymbol{K}\right)_{\mathrm{XX}}\right) \tag{58}$$

$$p_{16} = \ln\frac{\left(\boldsymbol{M}^{-1}\boldsymbol{K}\right)_{\mathrm{YY}}}{\left(\boldsymbol{M}^{-1}\boldsymbol{K}\right)_{\mathrm{XX}}} = \ln\left(\left(\boldsymbol{M}^{-1}\boldsymbol{K}\right)_{\mathrm{YY}}\right) - \ln\left(\left(\boldsymbol{M}^{-1}\boldsymbol{K}\right)_{\mathrm{XX}}\right) \tag{59}$$

$$p_{17} = \ln\left(\left(\boldsymbol{M}^{-1}\boldsymbol{C}\right)_{\mathrm{XX}}\right) \tag{60}$$

$$p_{18} = \ln\frac{\left(\boldsymbol{M}^{-1}\boldsymbol{C}\right)_{\mathrm{YY}}}{\left(\boldsymbol{M}^{-1}\boldsymbol{C}\right)_{\mathrm{XX}}} = \ln\left(\left(\boldsymbol{M}^{-1}\boldsymbol{C}\right)_{\mathrm{YY}}\right) - \ln\left(\left(\boldsymbol{M}^{-1}\boldsymbol{C}\right)_{\mathrm{XX}}\right) \tag{61}$$

Then,

$$\boldsymbol{M}^{-1}\boldsymbol{K} = \begin{bmatrix} \mathrm{e}^{p_{15}} & 0 \\ 0 & \mathrm{e}^{p_{15}+p_{16}} \end{bmatrix} \qquad \boldsymbol{M}^{-1}\boldsymbol{C} = \begin{bmatrix} \mathrm{e}^{p_{17}} & 0 \\ 0 & \mathrm{e}^{p_{17}+p_{18}} \end{bmatrix} \tag{62}$$

The phase was represented by the orientation of the tip of a tooth at time $t_{\mathrm{up}}(0)$:

$$p_{19} = \theta_1\left(0, t_{\mathrm{up}}(0)\right) \tag{63}$$

since the orientation of the tool at time $t_{\mathrm{up}}(0)$ should be consistent regardless of the other parameters, as long as the tachometer position and the mark on the tool detected by the tachometer are consistent.

During cutting, the values of $\boldsymbol{M}^{-1}\boldsymbol{K}_{\mathrm{c}}(t)$ and $\boldsymbol{M}^{-1}\boldsymbol{f}_0(t)$ are also necessary. Dividing through by the first element in the mass matrix reduces the number of parameters by one, as shown below. First, observe that $\boldsymbol{M}^{-1}$ can be written as

$$\boldsymbol{M}^{-1} = \begin{bmatrix} \frac{1}{\boldsymbol{M}_{\mathrm{XX}}} & 0 \\ 0 & \frac{1}{\boldsymbol{M}_{\mathrm{YY}}} \end{bmatrix} = \frac{1}{\boldsymbol{M}_{\mathrm{XX}}}\begin{bmatrix} 1 & 0 \\ 0 & \frac{\boldsymbol{M}_{\mathrm{XX}}}{\boldsymbol{M}_{\mathrm{YY}}} \end{bmatrix} \tag{64}$$

So, substituting Eqs. (25) and (26),

$$\boldsymbol{M}^{-1}\boldsymbol{K}_{\mathrm{c}}(t) = \begin{bmatrix} 1 & 0 \\ 0 & \frac{\boldsymbol{M}_{\mathrm{XX}}}{\boldsymbol{M}_{\mathrm{YY}}} \end{bmatrix} \sum_{k=1}^{n_{\mathrm{teeth}}} g_k(t) \begin{bmatrix} -\frac{K_{\mathrm{t}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{sc} - \frac{K_{\mathrm{n}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{s^2} & -\frac{K_{\mathrm{t}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{c^2} - \frac{K_{\mathrm{n}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{sc} \\ \frac{K_{\mathrm{t}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{s^2} - \frac{K_{\mathrm{n}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{sc} & \frac{K_{\mathrm{t}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{sc} - \frac{K_{\mathrm{n}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{c^2} \end{bmatrix}$$

$$\boldsymbol{M}^{-1}\boldsymbol{f}_0(t) = \begin{bmatrix} 1 & 0 \\ 0 & \frac{\boldsymbol{M}_{\mathrm{XX}}}{\boldsymbol{M}_{\mathrm{YY}}} \end{bmatrix} \sum_{k=1}^{n_{\mathrm{teeth}}} g_k(t) \left(h \begin{bmatrix} -\frac{K_{\mathrm{t}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{sc} - \frac{K_{\mathrm{n}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{s^2} \\ \frac{K_{\mathrm{t}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{s^2} - \frac{K_{\mathrm{n}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{sc} \end{bmatrix} + \begin{bmatrix} -\frac{K_{\mathrm{te}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{c} - \frac{K_{\mathrm{ne}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{s} \\ \frac{K_{\mathrm{te}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{s} - \frac{K_{\mathrm{ne}}}{\boldsymbol{M}_{\mathrm{XX}}}\overline{c} \end{bmatrix} \right)$$

In other words, dividing the pressure and edge coefficients by $\boldsymbol{M}_{\mathrm{XX}}$ allows for $\boldsymbol{M}_{\mathrm{XX}}$ to be eliminated as a parameter. As with the stiffness and damping, many of the parameters are known to be positive or to have a fairly consistent ratio, so it is beneficial to use logarithms and ratios as the parameters:

$$p_{20} = \ln \frac{K_{\mathrm{t}}}{\boldsymbol{M}_{\mathrm{XX}}} \tag{65}$$

$$p_{21} = \ln \frac{K_{\mathrm{n}}}{K_{\mathrm{t}}} = \ln \frac{K_{\mathrm{n}}/\boldsymbol{M}_{\mathrm{XX}}}{K_{\mathrm{t}}/\boldsymbol{M}_{\mathrm{XX}}} = \ln \frac{K_{\mathrm{n}}}{\boldsymbol{M}_{\mathrm{XX}}} - \ln \frac{K_{\mathrm{t}}}{\boldsymbol{M}_{\mathrm{XX}}} \tag{66}$$

$$p_{22} = \ln \frac{K_{\mathrm{te}}}{\boldsymbol{M}_{\mathrm{XX}}} \tag{67}$$

$$p_{23} = \ln \frac{K_{\mathrm{ne}}}{K_{\mathrm{te}}} = \ln \frac{K_{\mathrm{ne}}/\boldsymbol{M}_{\mathrm{XX}}}{K_{\mathrm{te}}/\boldsymbol{M}_{\mathrm{XX}}} = \ln \frac{K_{\mathrm{ne}}}{\boldsymbol{M}_{\mathrm{XX}}} - \ln \frac{K_{\mathrm{te}}}{\boldsymbol{M}_{\mathrm{XX}}} \tag{68}$$

$$p_{24} = \ln \frac{\boldsymbol{M}_{\mathrm{YY}}}{\boldsymbol{M}_{\mathrm{XX}}} = -\ln \frac{\boldsymbol{M}_{\mathrm{XX}}}{\boldsymbol{M}_{\mathrm{YY}}} \tag{69}$$

Then,

$$\boldsymbol{M}^{-1}\boldsymbol{K}_{\mathrm{c}}(t) = \begin{bmatrix} 1 & 0 \\ 0 & \mathrm{e}^{-p_{24}} \end{bmatrix} \sum_{k=1}^{n_{\mathrm{teeth}}} g_k(t) \begin{bmatrix} -\mathrm{e}^{p_{20}}\overline{sc} - \mathrm{e}^{p_{20}+p_{21}}\overline{s^2} & -\mathrm{e}^{p_{20}}\overline{c^2} - \mathrm{e}^{p_{20}+p_{21}}\overline{sc} \\ \mathrm{e}^{p_{20}}\overline{s^2} - \mathrm{e}^{p_{20}+p_{21}}\overline{sc} & \mathrm{e}^{p_{20}}\overline{sc} - \mathrm{e}^{p_{20}+p_{21}}\overline{c^2} \end{bmatrix}$$

$$\boldsymbol{M}^{-1}\boldsymbol{f}_0(t) = \begin{bmatrix} 1 & 0 \\ 0 & \mathrm{e}^{-p_{24}} \end{bmatrix} \sum_{k=1}^{n_{\mathrm{teeth}}} g_k(t) \left(p_3 \begin{bmatrix} -\mathrm{e}^{p_{20}}\overline{sc} - \mathrm{e}^{p_{20}+p_{21}}\overline{s^2} \\ \mathrm{e}^{p_{20}}\overline{s^2} - \mathrm{e}^{p_{20}+p_{21}}\overline{sc} \end{bmatrix} + \begin{bmatrix} -\mathrm{e}^{p_{22}}\overline{c} - \mathrm{e}^{p_{22}+p_{23}}\overline{s} \\ \mathrm{e}^{p_{22}}\overline{s} - \mathrm{e}^{p_{22}+p_{23}}\overline{c} \end{bmatrix} \right)$$

In summary, $\boldsymbol{p} = \begin{bmatrix} p_1 & \cdots & p_{24} \end{bmatrix}^{\mathsf{T}}$ are all the parameters needed for the milling dynamics model, Eq. (4). Note that when fitting to time series, the initial conditions are also unknowns which need to be estimated.

4.2. Overview of trajectory matching

One way to estimate the parameters of a dynamics model is to solve an optimization problem for the parameters, where the objective function describes how closely trajectories predicted by the model match the measured time series. The objective function incorporates any prior belief about the parameters and penalizes differences between the noisy measured time series and the noiseless measurements corresponding to the predicted trajectories. This works even if the measurements represent only a subset of the state variables, such as positions but not velocities. For cases where the measurement noise vectors are independent, additive, and multivariate normal and the prior belief on the parameters is a multivariate normal distribution independent of the measurement noise, the optimization problem can be expressed as a nonlinear least squares problem which accounts for the distributions of the measurement noise and the prior belief. This approach is used to estimate parameters in Sections 4.5 and 4.6.

Fully describing a trajectory with the dynamics model requires the initial conditions of the trajectory in addition to the model parameters. Typically, these initial conditions are unknown and, as a result, need to be estimated simultaneously with the model parameters in order to match the predicted trajectory to the measurements. For an ODE model, these initial conditions consist of the system state at a single instant in time. For a DDE model with delay τ, however, the initial conditions consist of the trajectory of $\boldsymbol{x}(t)$ over a continuous interval of length τ, due to the dependency of the dynamics on past values of $\boldsymbol{x}(t)$. The spectral element method provides a convenient approximate representation of these initial conditions: the approximation of $\boldsymbol{x}(t)$ at the nodes in the first segment. By representing the initial conditions in this way, the future trajectory can easily be approximated using the spectral element method, as described in Section 2.3.

In summary, the parameters of a dynamics model can be estimated from time series by matching trajectories predicted by the model to the time series measurements. Under certain assumptions, this can be framed as a nonlinear least squares optimization problem which accounts for the covariance of the measurement noise and any prior belief on the parameters. The unknowns to optimize consist of both the unknown model parameters and the unknown initial conditions. The unknown model parameters may consist of parameters which are common to multiple time series, as well as parameters which are separate for each time series. Regardless, all of the unknowns can be concatenated together into a single vector of unknowns for the optimization problem. The optimization problem can be solved using standard techniques. For more details on the objective function, see Appendix A.

4.3. Estimating the spindle speed

As described in Section 3, the milling machine was instrumented with a tachometer to more accurately estimate the spindle speed. This section describes how the spindle speed was estimated from the tachometer signal. The tachometer signal was approximately a square wave between 0 V and 5 V; the edges of the signal were used to estimate the spindle speed. First, the upward and downward edges were estimated by finding the intersections between the tachometer signal and a threshold of 2.5 V, using linear interpolation between adjacent points of the tachometer signal. This is illustrated in Fig. 6. After estimating the edges, a linear model

$$t_{\mathrm{up}}(i) = t_{\mathrm{up}}(0) + Ti \tag{70}$$

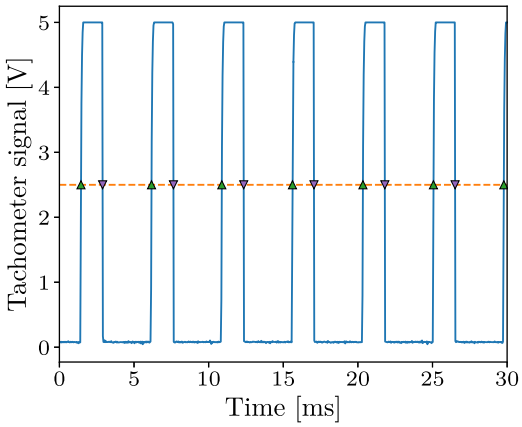


Fig. 6. Example of finding the upward and downward edges of the tachometer signal; only the first 30 ms is shown, due to space limitations. The tachometer signal is the solid blue line; the threshold of 2.5 V shown with the dashed orange line; the times corresponding to the detected upward and downward edges are indicated with upward-pointing and downward-pointing triangular markers, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

was fit to the indices and values of the estimated upward edges using ordinary least squares (OLS), where the first upward edge is $t_{\text{up}}(0)$, the next upward edge is $t_{\text{up}}(1)$, etc. The estimated spindle speed was given by $\Omega = 2\pi/T$. Finally, the estimated values of $t_{\text{up}}(0)$ and T were checked in various ways to verify that they were a good fit: (1) the estimated Ω was checked against the nominal spindle speed reported by the milling machine, (2) the residuals between the actual estimated upward edges and Eq. (70) were verified to be no more than a threshold fraction of T, and (3) the relative errors from T for the intervals between consecutive upward edges and consecutive downward edges were verified to be less than a threshold. If any of these checks failed, then the experiment was thrown out. The most common cause of failure was a small chip from the workpiece getting stuck to the side of the tool, which would occasionally interfere with the tachometer measurements.

Eq. (70) was fit separately to the edges from before cutting started and the edges during cutting, since the spindle speed changed slightly during cutting due to the additional load on the machine. The estimated spindle speed before cutting was used to estimate the parameters of steady-state vibration before cutting started (Section 4.4), and the estimated spindle speed during cutting was used for fitting the dynamics model (Sections 4.5 and 4.6). Additionally, the estimate of $t_{\text{up}}(0)$ for the data during cutting was used to match the phases of the time series, since the orientation of the tool should have been the same for all upward edges.

4.4. Estimating steady-state vibration before cutting started

For each experiment, the machine was started with the tool outside the workpiece and allowed to reach steady-state, then data collection was initialized, and then the tool was fed into the workpiece. The steady-state motion of the tool before cutting was estimated for two reasons: (1) the covariance of the difference between the measurements and the estimated motion was a good estimate of the covariance of the measurement noise, and (2) the estimated motion was a good initial guess for $\boldsymbol{q}_{\text{ss}}(t)$ (Eq. (49)) for use in fitting the vibration between cuts (Section 4.5).

For each time series, the steady-state motion was estimated by fitting a Fourier series

$$f(t) = \frac{a_0}{2} + \sum_{k=1}^{n} \left(a_k \cos k\Omega t + b_k \sin k\Omega t \right) \tag{71}$$

to each of the two position components of the measurements separately, using OLS. The value of Ω had been estimated earlier from the tachometer signal, as described in Section 4.3. For estimating $\boldsymbol{q}_{\text{ss}}(t)$, the Fourier series were limited to $n = 1$; see Fig. 7 for an example. For estimating the covariance of the measurement noise, $n = 10$ was used; the additional terms allowed fitting the data slightly more closely to obtain a slightly better estimate of the noise covariance in case the steady-state vibration did not perfectly match Eq. (49). Even if the noise is clearly noticeable in the data, as in Fig. 7, this method still estimates $\boldsymbol{q}_{\text{ss}}(t)$ well.

4.5. Fitting vibration between cuts and estimating the phase of cuts

For the low radial immersion experiments, most of the time was spent with the tool between cuts, i.e. with no teeth in contact with the workpiece. So, the model could be fit to the data between cuts with fewer parameters; the parameters describing the cutting forces were not necessary. In particular, Eq. (32) could be fit to the data instead of Eq. (4). By fitting fewer model parameters, the optimization algorithm converges faster and is more likely to find the global optimum. As an additional benefit, fitting the vibration between cuts can be used to estimate the phase when cuts occur. This section describes how the phase of the cuts and the relevant model parameters can be estimated.

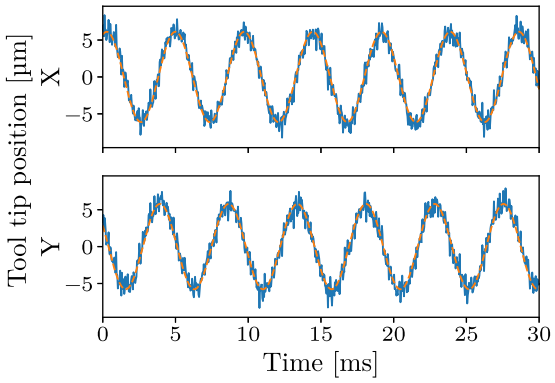


Fig. 7. Example of fitting to the observations before cutting started; only the first 30 ms is shown, due to space limitations. The components in the X and Y directions are shown on separate axes. The position measurements are the solid blue lines, and the corresponding fitted steady-state vibration $\boldsymbol{q}_{ss}(t)$ is indicated with dashed orange lines. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

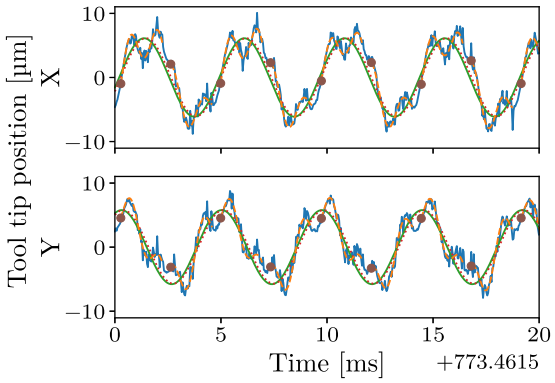


Fig. 8. Example of fitting to the observations between cuts. Note that only part of one time series is shown; the model is actually fit to multiple time series simultaneously. The components in the X and Y directions are shown on separate axes. The position measurements are the solid blue lines, and the corresponding fitted positions $\boldsymbol{q}(t)$ between cuts are indicated with dashed orange lines. The estimated initial condition for each interval between cuts is indicated with a brown circular marker. The steady-state response $\boldsymbol{q}_{ss}(t)$ estimated from the data before cutting started (Section 4.4) and the updated estimate of $\boldsymbol{q}_{ss}(t)$ from the data between cuts (Section 4.5) are shown with solid green and dotted red lines, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Observe that if a time interval is between cuts, then all intervals which are integer multiples of τ later in time are also between cuts, since cuts are separated in time by τ. Given the information obtained in Sections 4.3 and 4.4, intervals between cuts can be described by two values: the interval length $\alpha\tau$, and the time $\varsigma\tau$ between the first upward tachometer edge $t_{up}(0)$ and the start of an interval between cuts. In other words, the intervals described by these two values are $[t_{up}(0) + \varsigma\tau + k\tau,\ t_{up}(0) + \varsigma\tau + k\tau + \alpha\tau]$ for $k = 0, 1, \ldots$.

If suitable values of ς and α were known, then Eq. (32) could be fit to the data in the intervals described by ς and α using trajectory matching. The parameters to estimate are the relevant unknown model parameters which are common to all time series, i.e. $p_{15}, \ldots, p_{18}$, the relevant unknown model parameters which differ between time series, i.e. $p_9, \ldots, p_{14}$, and the initial condition at the start of each interval. Note that good initial guesses for $p_9, \ldots, p_{14}$ for each time series are available from fitting the data before cutting starts, as described in Section 4.4. The trajectory matching can be framed as a least squares optimization problem, as described in Section 4.2, using the analytical solution (Eq. (34)) for the predicted trajectories. The optimization problem can be solved using standard nonlinear least-squares methods, such as the `scipy.optimize.least_squares` SciPy library function [67] with the Trust Region Reflective algorithm. Only a subset of the parameters affects each term in the objective function, so sparse Jacobians can be used to significantly speed up the optimization algorithm. Specifically, each initial condition affects only the terms in the corresponding interval, and each time series has different values of $p_9, \ldots, p_{14}$. Fig. 8 shows an example of fitting the model to the data in the intervals between cuts. The fitted trajectories in the intervals are indicated with dashed orange lines, and the estimated initial condition for each interval is indicated with a brown circular marker.

Unfortunately, the phase of the intervals between cuts was unknown, since none of the sensors could directly detect contact between the tool and the workpiece. So, it was necessary to find suitable values of ς and α using the time series data. One way to do this is to take advantage of the fact that Eq. (32) should fit the data in intervals between cuts more closely than data in intervals which contain cuts, since the cutting forces cause the motion of the tool during cuts to deviate from the dynamics described

by Eq. (32). So, given a set of candidate values of (ς, α), the best candidate from the set can be chosen by comparing the quality of the fits for these candidates, as described by the least squares objective function. The question, then, is how to strategically choose candidates for (ς, α) in order to narrow in on the correct values.

First, it is useful to determine the maximum possible value of α for which there exists a value of ς such that all the intervals described by ς and α are between cuts. This maximum value for α is $1-\gamma$, where γ is the fraction of each tooth passage period for which a tooth may be cutting in any of the time series when they are aligned by their respective values of $t_{\text{up}}(0)$. When dealing with multiple time series of differing radial immersions and axial cut depths,

$$\gamma = \frac{\left(\max\limits_k \left(\theta_{\text{ex},k} + \theta_{\text{lag},k}\right)\right) - \left(\min\limits_k \theta_{\text{en},k}\right)}{2\pi/n_{\text{teeth}}} \tag{72}$$

where $\theta_{\text{en},k}$, $\theta_{\text{ex},k}$, and $\theta_{\text{lag},k}$ are the tooth tip cut entry, tooth tip cut exit, and lag angles for the time series with index k; these angles are described in more detail in Section 2.1. The process of estimating the phase can be implemented as a sequence of steps, where each step chooses between a small number of candidates of (ς, α). The first step starts with a smaller value of α in order to find a value of ς which corresponds to intervals between cuts, and then subsequent steps increase α and adjust the value of ς. After multiple steps, α approaches the maximum value of $1-\gamma$. At that point, the intervals described by ς and α are the largest possible intervals between cuts, and their endpoints describe when cuts occur.

The first step considers four different candidates: $\varsigma \in \left\{\frac{0}{4}, \frac{1}{4}, \frac{2}{4}, \frac{3}{4}\right\}$, all with $\alpha = \frac{3}{4} - \gamma$. This value for α is chosen so that an interval of length $\gamma\tau$ cannot simultaneously overlap the intervals described by all four candidates. This ensures that there is at least one candidate which represents intervals which are entirely between cuts. Fig. 9(a) shows how the choice of $\alpha = \frac{3}{4} - \gamma$ ensures this property. In this figure, the intervals for each of the four candidates are indicated with a solid color, and possible cutting intervals of duration $\gamma\tau$, separated by time τ, are indicated with hatched rectangles. If the hatched rectangles are slid together across the timeline, they never intersect all four candidates simultaneously. Fig. 9(c) illustrates the same thing, but is wrapped around a circle to more easily show the periodicity. Once the model has been fit to the intervals for each candidate value of ς, the value of ς for which the model best fits the data is selected. Let ς_0 and α_0 be these initially-selected values.

In the following steps, the selected intervals are repeatedly expanded, until α is as close as desired to $1-\gamma$. For each step k, given values of ς_{k-1} and α_{k-1} which were selected on the previous step, ς_k and α_k can be chosen by considering two candidates which overlap the previously-selected candidate but are somewhat larger. This is illustrated in Figs. 9(b) and 9(d). The previously-selected candidate with intervals of length $\alpha_{k-1}\tau$ is indicated with a dotted background, and the two expanded candidates with intervals of length $\alpha_k\tau$ are indicated with solid backgrounds. If $\alpha_k = \alpha_{k-1} + \frac{1}{2}(1-\gamma-\alpha_{k-1})$, then if the hatched regions are slid together across the timeline, they cannot overlap both candidates simultaneously while remaining outside the intervals specified by $(\varsigma_{k-1}, \alpha_{k-1})$. So, if $(\varsigma_{k-1}, \alpha_{k-1})$ represents intervals which are entirely between cuts, then at least one of the candidates is guaranteed to represent intervals which are entirely between cuts. As the number of steps increases, α rapidly approaches the maximum of $1-\gamma$, which can be shown as follows. The value of α_i is given by

$$\alpha_i = \begin{cases} \frac{3}{4} - \gamma & \text{if } i = 0 \\ \alpha_{i-1} + \frac{1}{2}(1-\gamma-\alpha_{i-1}) & \text{if } i \in \{1, 2, \ldots\} \end{cases} \tag{73}$$

So, the difference between $1-\gamma$ and α_i is given by

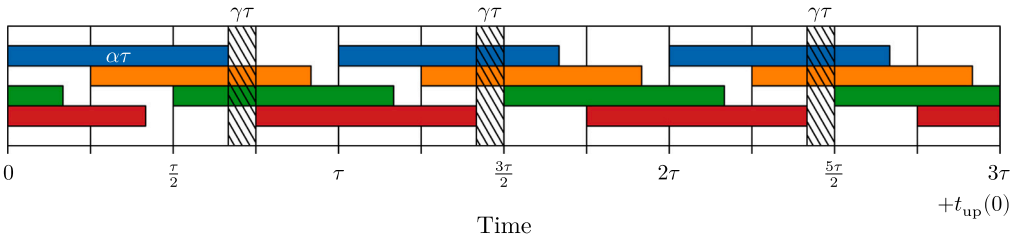
$$(1-\gamma) - \alpha_i = \begin{cases} (1-\gamma) - \left(\frac{3}{4} - \gamma\right) & \text{if } i = 0 \\ (1-\gamma) - \left(\alpha_{i-1} + \frac{1}{2}(1-\gamma-\alpha_{i-1})\right) & \text{if } i \in \{1, 2, \ldots\} \end{cases} \tag{74}$$

$$= \begin{cases} \frac{1}{4} & \text{if } i = 0 \\ \frac{1}{2}(1-\gamma-\alpha_{i-1}) & \text{if } i \in \{1, 2, \ldots\} \end{cases} \tag{75}$$

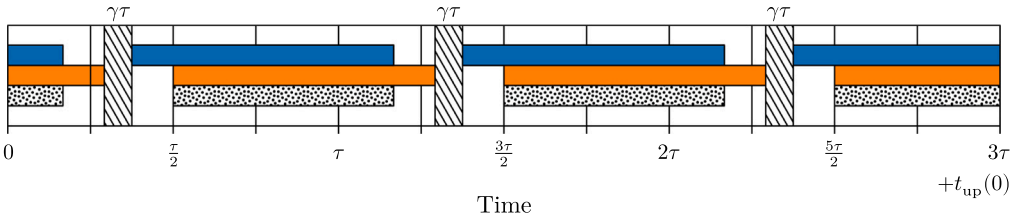
$$= \frac{1}{4}\left(\frac{1}{2}\right)^i = 2^{-(i+2)} \tag{76}$$

In other words, the difference between $1-\gamma$ and α exponentially decays towards zero. For example, after five expansion steps, $(1-\gamma) - \alpha_5 < 1\%$. Using this procedure, the only limitations on the precision of the estimate of ς are the time step between measurements and the uncertainty due to measurement noise.

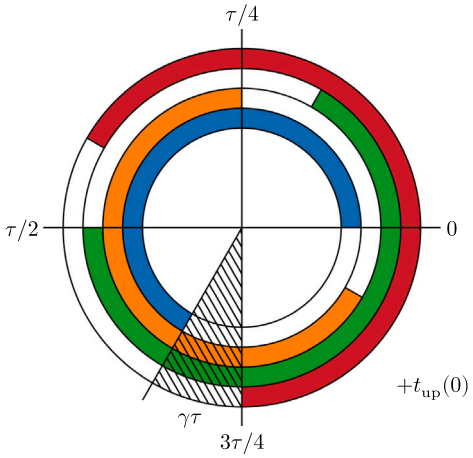
Once ς has been estimated using this procedure, it can be used in combination with $t_{\text{up}}(0)$ to specify the times when cuts occur, for fitting the milling model as described in the following section. While the estimate of ς obtained from this procedure is usually fairly close to the correct value, measurement noise can lead to larger error than would be ideal. The following section adds a parameter to allow for small errors in the estimates of $t_{\text{up}}(0)$ and ς. To allow for larger errors, it can be beneficial to also try values of ς somewhat farther away, selecting the value for which the model best fits the data. The estimates of the model parameters $p_9, \ldots, p_{18}$, obtained while fitting Eq. (32) to the data for the chosen value of ς, are used in the following section to fit the remaining parameters.



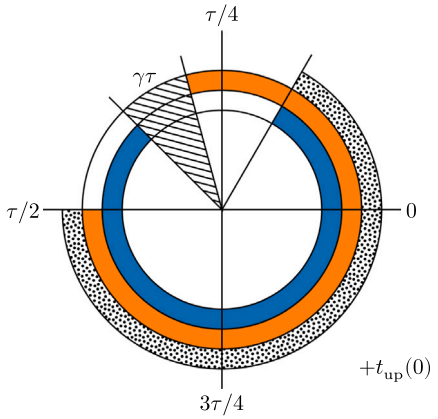
(a) Obtaining an initial estimate of an interval between cuts, using $\alpha = 3/4 - \gamma$.



(b) Expanding an estimate of an interval between cuts.



(c) Figure 9a wrapped around a circle, to illustrate the periodicity.



(d) Figure 9b wrapped around a circle, to illustrate the periodicity.

Fig. 9. Strategy for estimating the phase of the cuts by the tool. The intervals to try fitting as vibration between cuts are indicated with solid backgrounds; the size of the interval when the tool may be cutting is indicated with a hatched region; and the previously chosen interval being expanded in Figs. 9(b) and 9(d) is indicated with a dotted background.

4.6. Estimating the cutting coefficients and mass asymmetry

After completing the stages described in the previous sections, the only remaining parameters which have not been estimated are those describing the cutting coefficients and mass asymmetry, i.e. $p_{20}, \ldots, p_{24}$. These parameters can be estimated by fitting the full milling model, i.e. Eq. (4), to the time series after cutting started, using the trajectory matching approach described in Section 4.2 with the spectral element method described in Section 2.3. This is illustrated in Fig. 10.

Most of the parameter values estimated in the previous sections were used as-is. The only exception was an additional parameter for each time series to allow for small errors in the estimates of parameters p_8 and p_{19}, since small errors in the phase of the cuts can significantly affect the quality of the fit to the time series. To allow for these small errors, the orientation of the tip of tooth 1 at the estimate of $t_{\text{up}}(0)$ was represented by a shifted logistic curve:

$$\theta_1(0, \tilde{p}_8) = \tilde{p}_{19} - 0.05 \left(\frac{2\pi}{n_{\text{teeth}}} \right) \left(\frac{2}{1 + (1 + \mathrm{e}^{-p_{25}})} - 1 \right) \tag{77}$$

where $\tilde{p}_8$ is the value of p_8 estimated in Section 4.3, $\tilde{p}_{19}$ is the value of p_{19} estimated in Section 4.5, and p_{25} is an additional parameter for each time series.

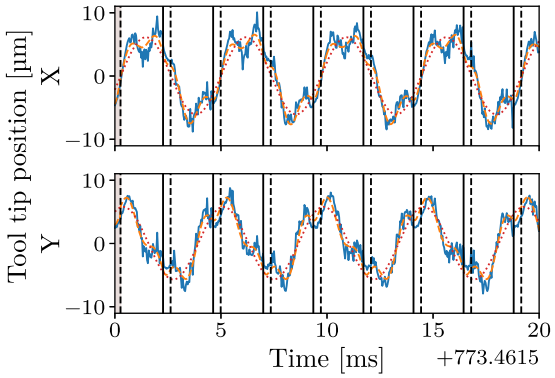


Fig. 10. Example of fitting using the complete milling model (Eq. (4)). Note that only part of one time series is shown; the model is actually fit to multiple time series simultaneously. The components in the X and Y directions are shown on separate axes. The position measurements are the solid blue lines, and the corresponding fitted positions $\boldsymbol{q}(t)$ are indicated with dashed orange lines. The start and end of each cut are indicated with solid and dashed vertical lines, respectively. The cut containing the explicitly parameterized nodes is indicated with a shaded background. The steady-state response $\boldsymbol{q}_{ss}(t)$ estimated from the data between cuts (Section 4.5) is shown with a dotted red line. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

To match the time series data, the initial conditions for each time series must be estimated simultaneously with the unknown model parameters. The initial conditions for a DDE with delay τ consist of the states $\boldsymbol{x}$ over an interval of length τ. For the approach presented here, this interval of states was represented by state values at the spectral element nodes for one cut. Specifically, the first spectral element segment was explicitly represented by the position $\boldsymbol{q}$ at each spectral element node in the segment, the velocity $\dot{\boldsymbol{q}}$ at the first node in the segment, and the velocity at the last node in the segment. The velocities between the first and last nodes were not necessary, because they affected neither the position measurements in the first segment nor the states of later segments. An initial guess for the explicitly represented nodes was obtained by fitting a polynomial of order $\left(\min\left\{n-1, \lfloor\sqrt{n}\rfloor\right\}\right)$ to the n time series measurements within the initial cut. Given the representation of the initial segment and the model parameters, an estimate of the state $\boldsymbol{x}$ at any time in the time series could be computed using the spectral element method, as described in Section 2.3. So, the model parameters and initial conditions could be estimated by finding the values of the parameters and initial conditions for which the computed trajectory most closely matched the time series data, as described in Section 4.2.

For the solving the least squares optimization problem to fit the time series, the `least_squares` SciPy library function [67] with the Trust Region Reflective algorithm was used. As in Section 4.5, only a subset of the parameters affected each term in the objective function, so sparse Jacobians were used to significantly speed up the optimization algorithm. Specifically, each initial condition segment and each parameter p_{25} affected only the terms in the corresponding time series. Once the parameters were estimated, the milling model could be used to make predictions. For example, the spectral element method was used to generate a stability chart for different spindle speeds and cutting depths.

5. Results and discussion

The methods presented in this article were designed so that a machine operator could, in a few minutes, conduct a few short, low radial immersion test cuts; run the automated parameter estimation procedure; and then automatically generate a stability chart, like the one shown in Fig. 11, using the estimated parameters. To demonstrate their applicability to a real milling system, the methods presented in this article were applied to time series collected using the experimental setup described in Section 3.

Given a set of a few time series, the methods described in Section 4 were used to estimate the parameters of the dynamics model described in Section 2. Then, using these parameter estimates, the spectral element method described in Section 2.3 and Appendix B was used to compute approximate CMs over a grid of spindle speeds and axial cut depths. Each point in the grid was predicted to be stable if the magnitudes of all the CMs were less than one, and unstable otherwise. This procedure of estimating the parameters from a set of time series and predicting the stability over a grid of points was repeated for various sets of time series, in order to provide an indication of the variability in the predictions. Specifically, it was repeated for each subset of four unique time series from a set of five time series. The spindle speeds and axial cut depths for the five time series are indicated by black boxes in Fig. 11. Note that all of these cases have low axial cut depths. Low axial cut depths are a good choice for test cuts in order to minimize the risk of damaging the tool, since the system is more likely to be stable, and even if it is unstable, the cutting forces are smaller.

The grids of spectral element stability predictions corresponding to the subsets of time series were overlaid on each other to form the background color in Fig. 11: the background color of each point indicates the fraction of the subsets of the time series for which the point was predicted to be stable. For comparison, true stability assessments obtained experimentally [2] are shown with triangular, cross-shaped, and circular markers. There is noticeable variability in the stability predictions, as indicated by the width of the transition between the background colors corresponding to the stable and unstable regions. However, they match the

Table 1
Comparison of values obtained from separate modal and cutting force tests [2] to the values estimated from time series using the methods described in Section 4. For the separate test values, only the nominal values from [2] are shown; the uncertainty was unavailable. The estimated values are reported as means and standard deviations over the data subsets.

Expression	Description	Separate tests	Estimated	Units
$\sqrt{(\boldsymbol{M}^{-1}\boldsymbol{K})_{\mathrm{XX}}}$	Angular natural frequency, XX	4.58×10^{3}	$(4.53 \pm 0.03) \times 10^{3}$	rad/s
$\sqrt{(\boldsymbol{M}^{-1}\boldsymbol{K})_{\mathrm{YY}}}$	Angular natural frequency, YY	4.57×10^{3}	$(4.53 \pm 0.01) \times 10^{3}$	rad/s
$\frac{(\boldsymbol{M}^{-1}\boldsymbol{C})_{\mathrm{XX}}}{2\sqrt{(\boldsymbol{M}^{-1}\boldsymbol{K})_{\mathrm{XX}}}}$	Damping ratio, XX	10.7×10^{-3}	$(7.37 \pm 2.83) \times 10^{-3}$	
$\frac{(\boldsymbol{M}^{-1}\boldsymbol{C})_{\mathrm{YY}}}{2\sqrt{(\boldsymbol{M}^{-1}\boldsymbol{K})_{\mathrm{YY}}}}$	Damping ratio, YY	9.96×10^{-3}	$(5.18 \pm 0.70) \times 10^{-3}$	
$K_{\mathrm{t}}/\boldsymbol{M}_{\mathrm{XX}}$	Normalized tangential cutting pressure	12.3×10^{9}	$(6.88 \pm 0.65) \times 10^{9}$	Pa/kg
$K_{\mathrm{n}}/\boldsymbol{M}_{\mathrm{XX}}$	Normalized normal cutting pressure	4.29×10^{9}	$(4.70 \pm 2.45) \times 10^{9}$	Pa/kg

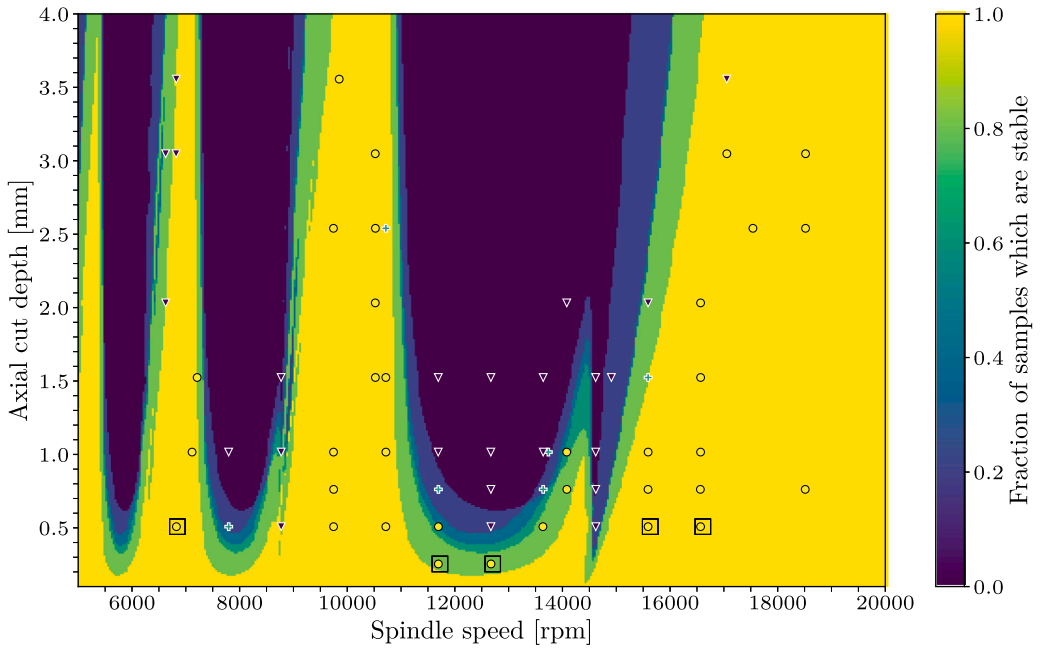


Fig. 11. Stability charts for the estimated parameter values. For each point (Ω, b), the background color indicates the fraction of subsets of the data for which the estimated parameters predicted that the point would be stable. For comparison, the experimental stability assessments for this system from [2] are indicated with markers (stable: yellow circle, borderline: green cross, unstable: purple triangle). The values of Ω and b for the time series used for fitting are indicated with black boxes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

true stability assessments fairly closely. Given a stability chart like this, the operator could easily select a good operating point by finding a point which meets the desired criteria and is a reasonable distance from the unstable regions.

While the purpose of the methods is to provide stability predictions, the parameter values themselves may also be of some interest. Table 1 compares some of the nominal values obtained from separate modal and cutting force tests [2] to the values estimated from the subsets of time series as described above. The estimated values are reported as means and standard deviations over the subsets of time series. Note that the separate tests had some uncertainty as well, but only the nominal values were available for comparison. Based on the table, the estimated natural frequencies were quite close to the nominal values. Similarly, the damping ratio and normalized cutting pressure values were close in order of magnitude to the nominal values. It is unclear how much of the differences were due to errors in the nominal values obtained from the separate tests or due to errors in the values estimated from the time series. Regardless, the estimated values were sufficiently accurate to generate a useful stability chart as shown in Fig. 11.

Also, while the purpose of the methods is to provide stability predictions via parameter estimation, comparing the fitted trajectories to the experimental measurements provides some useful insight. For the same five time series which were used to generate Fig. 11, the fitting procedure was run for all five time series together. For the fit of the data before cutting started (Section 4.4), Fig. 12 shows the estimated $\boldsymbol{q}_{\mathrm{ss}}(t)$ for the data before cutting started. The good fit confirms that a single sinusoid is sufficient to represent the steady-state vibration. For the fit of the data between cuts (Section 4.5), Fig. 13 shows the estimated $\boldsymbol{q}(t)$ between cuts and the updated estimate of $\boldsymbol{q}_{\mathrm{ss}}(t)$. The model for the vibration between cuts fits the data very well; this strongly supports the proposal of incorporating nonzero $\boldsymbol{f}_{\mathrm{u}}(t)$ into the model to handle cases where the tool is not perfectly centered about

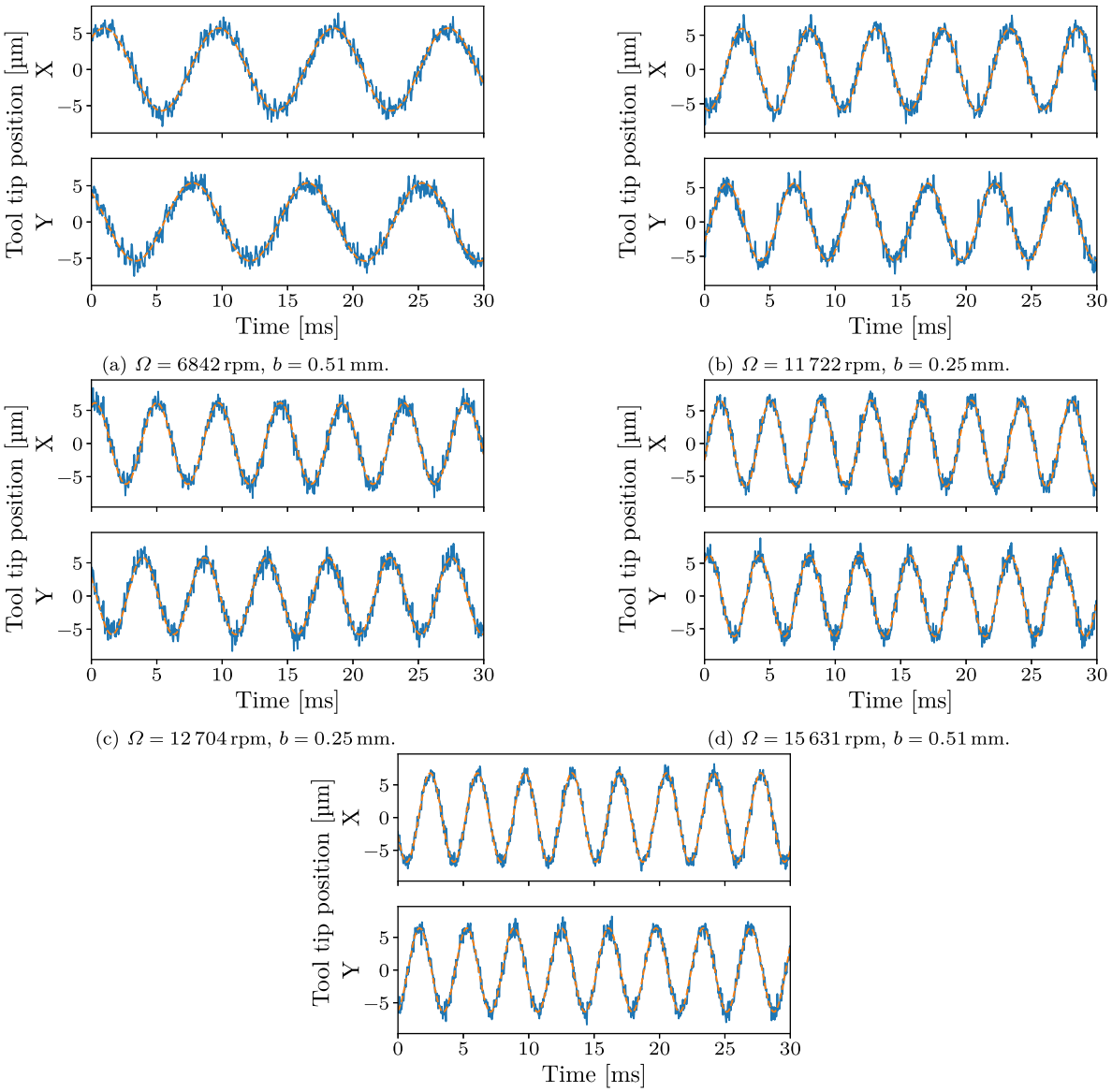
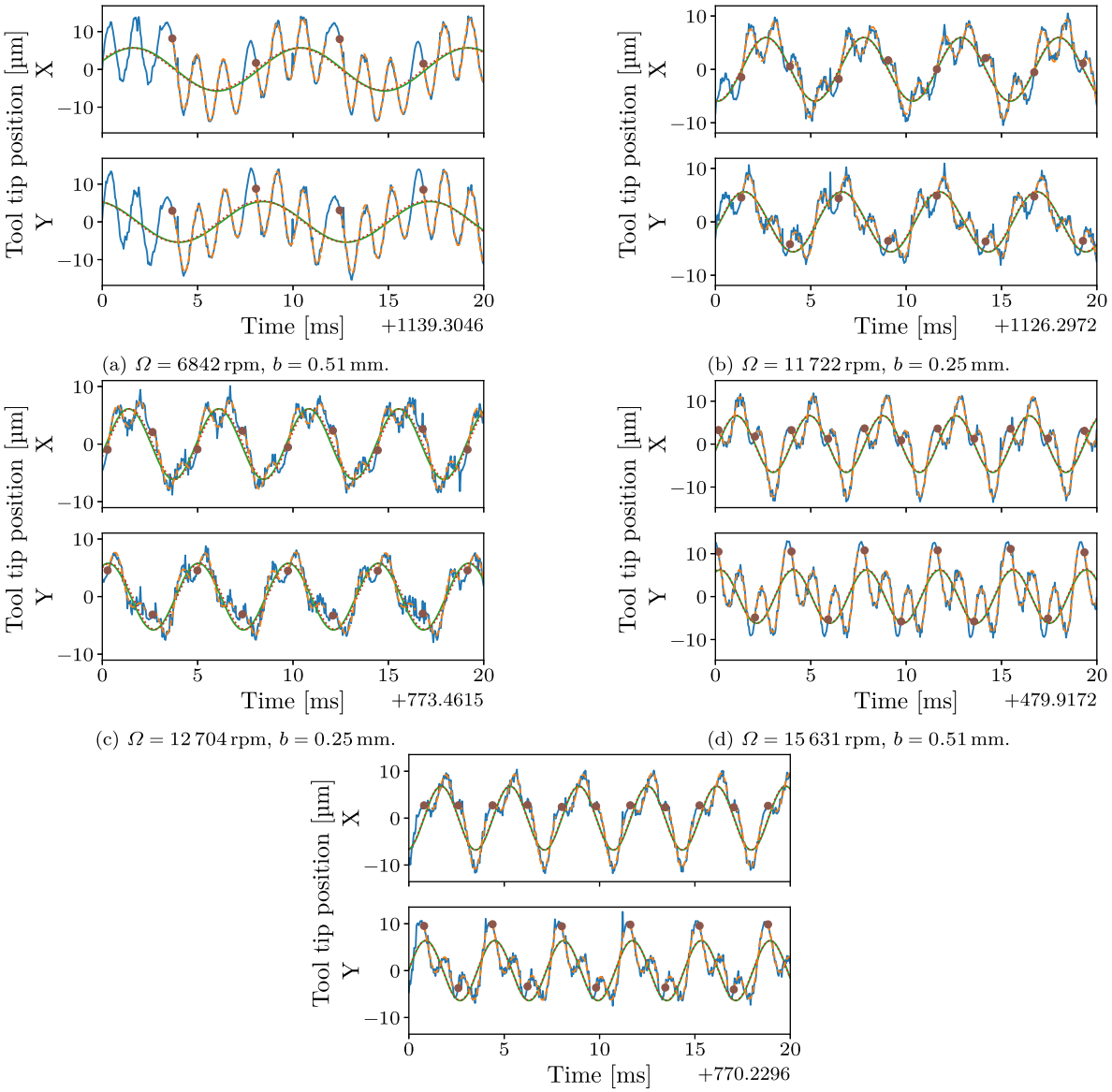


Fig. 12. Fit to the observations before cutting started; only the first 30 ms is shown, due to space limitations. The components in the X and Y directions are shown on separate axes. The noisy position measurements are the solid blue lines, and the corresponding fitted steady-state vibration $\boldsymbol{q}_{ss}(t)$ (Section 4.4) is indicated with dashed orange lines. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the axis of rotation. For fitting all the data during cutting (Section 4.6), Fig. 14 shows the estimated $\boldsymbol{q}(t)$. While the model fits the most of the time series fairly well, some of the errors between the observations and fitted state are systematic, i.e. not just due to measurement noise. This suggests that the DDE model describing the dynamics during each cut could be improved further or, possibly, some of the "known" parameter values ($p_1, \ldots, p_6$) were not quite correct. This also explains why some of the errors in the stability estimates in Fig. 11 appear to be systematic, rather than due to measurement noise.

In summary, using the proposed methods, it is possible to quickly collect data from a small number of low radial immersion test cuts, using sensors attached to a milling machine, and then generate a stability chart which can be used to select appropriate parameters in the stable region. The results demonstrate that the model fits the data fairly well, although there is room for further improvement. The stability chart in Fig. 11 is close enough to the correct stability boundary to be useful in practice. Fitting to subsets of the data provides an indication of the variability or uncertainty in the stability estimates. While interpreting the variability as



(a) $\Omega = 6842\,\text{rpm},\ b = 0.51\,\text{mm}.$

(b) $\Omega = 11\,722\,\text{rpm},\ b = 0.25\,\text{mm}.$

(c) $\Omega = 12\,704\,\text{rpm},\ b = 0.25\,\text{mm}.$

(d) $\Omega = 15\,631\,\text{rpm},\ b = 0.51\,\text{mm}.$

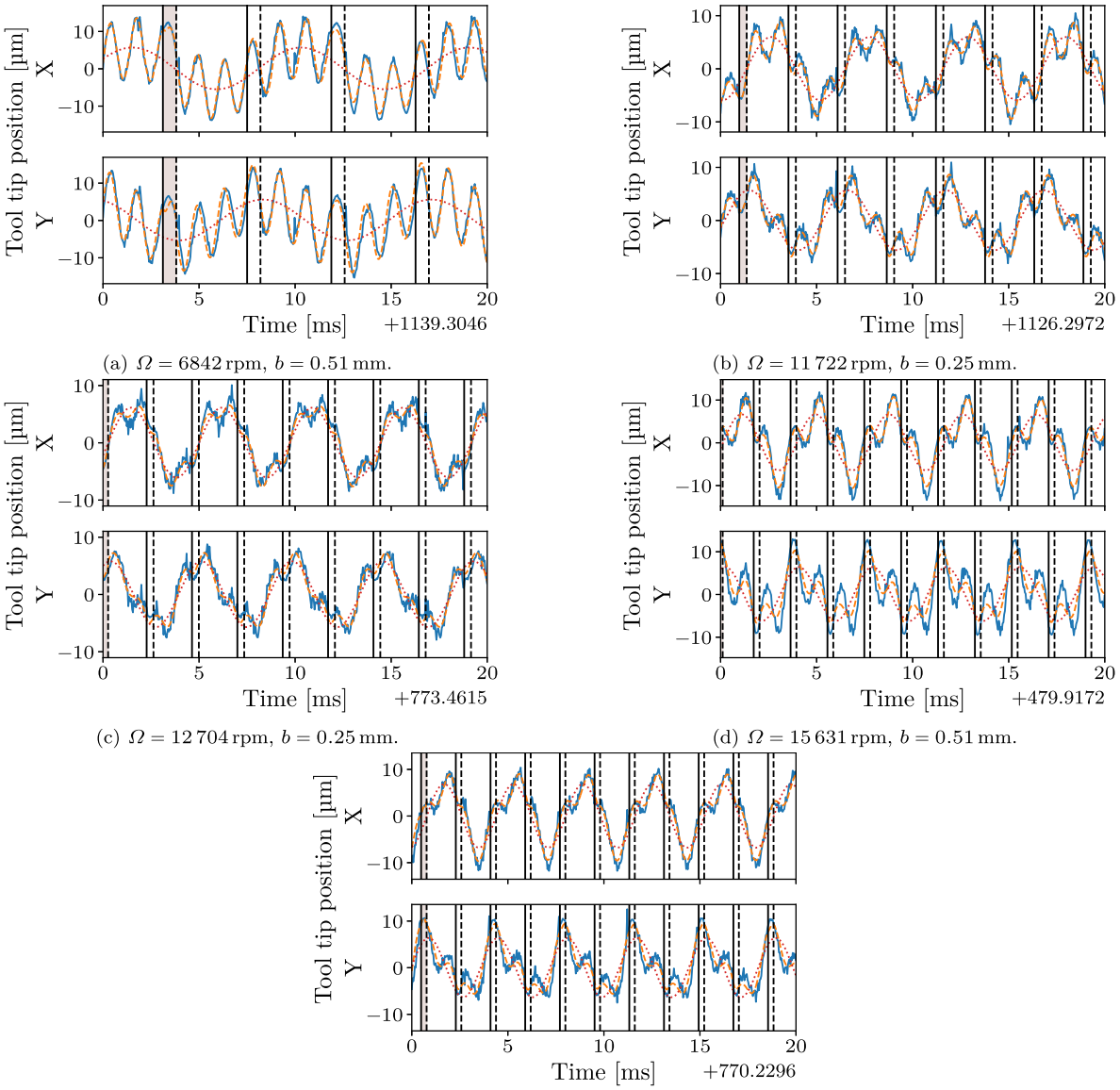
(e) $\Omega = 16\,609\,\text{rpm},\ b = 0.51\,\text{mm}.$

Fig. 13. Fit to the observations between cuts; only the first 10% of the data is shown, due to space limitations. The components in the X and Y directions are shown on separate axes. The position measurements are the solid blue lines, and the corresponding fitted positions $\boldsymbol{q}(t)$ between cuts (Section 4.5) are indicated with dashed orange lines. The estimated initial condition for each interval between cuts is indicated with a brown circular marker. The steady-state response $\boldsymbol{q}_{\text{ss}}(t)$ estimated from the data before cutting started (Section 4.4) and the updated estimate of $\boldsymbol{q}_{\text{ss}}(t)$ from the data between cuts (Section 4.5) are shown with solid green and dotted red lines, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

uncertainty may be somewhat misleading because the errors are partially systematic, the variability information allows the operator to make a more informed decision. Given the stability chart in Fig. 11, the operator could select a point a reasonable distance from the unstable regions which meets the operator's desired criteria.

6. Conclusion

Milling is one of the most widely used material removal processes. Despite this popularity, milling in its present state has a number of limitations resulting from vibrations of the cutting tool. Chatter, a vibration induced dynamic instability, is particularly undesirable as it causes surface location error, tool damage, and sub-optimal material removal rates. For these reasons, much research has focused on chatter, including modeling, numerical prediction, stability analyses, chatter detection, and vibration absorption.



(a) $\Omega = 6842$ rpm, $b = 0.51$ mm.

(b) $\Omega = 11\,722$ rpm, $b = 0.25$ mm.

(c) $\Omega = 12\,704$ rpm, $b = 0.25$ mm.

(d) $\Omega = 15\,631$ rpm, $b = 0.51$ mm.

(e) $\Omega = 16\,609$ rpm, $b = 0.51$ mm.

Fig. 14. Fit using the complete milling model (Eq. (4)); only the first 10% of the data is shown, due to space limitations. The components in the X and Y directions are shown on separate axes. The position measurements are the solid blue lines, and the corresponding fitted positions $\boldsymbol{q}(t)$ (Section 4.6) are indicated with dashed orange lines. The start and end of each cut are indicated with solid and dashed vertical lines, respectively. The cut containing the explicitly parameterized nodes is indicated with a shaded background. The steady-state response $\boldsymbol{q}_{ss}(t)$ estimated from the data between cuts (Section 4.5) is shown with a dotted red line. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

However, chatter research has failed to be extensively applied in industry, in part because of the challenges associated with system identification for milling models described by DDEs. Thus, this work sought to automatically identify model parameters from milling vibration time series, in order to enable industry to take advantage of existing research. The experimental setup was designed to closely resemble a common application by the aerospace industry—the deep pocket machining of monolithic aerospace structures.

This article described an automated method for estimating the parameters of a two-degree-of-freedom milling model from a small number of time series which could be collected using sensors attached to a milling machine. The milling model incorporated nonzero steady-state vibration of the milling tool, independent of cutting, to better match the motion of the tool in real time series. The parameters were strategically defined to be well-suited for estimation using optimization algorithms. The fitting method proceeded in stages which estimated a few parameters at a time in order to minimize the number of parameters which needed to be estimated simultaneously. Finally, the proposed method was evaluated on real experimental data. The results demonstrated that, despite some

limitations of the model, the estimated stability chart matched the true stability boundary fairly closely and provided valuable information which would allow an operator to select appropriate machining settings.

Although the parameters were estimated for the case of low radial immersion, several methods exist to use these same parameters to make stability predictions for higher radial immersions [36,68]. Additionally, note that using low radial immersion cutting tests for in-process identification is advantageous as it increases the frequency content of the excitation spectrum.

There are several possible extensions of the current work. For example, the long slender tools for this work commonly yield systems with a single dominant mode of vibration in each direction. A possible future extension is to consider tools with multiple modes of vibration in each direction. Additionally, further improvements to the models may allow them to fit the time series more closely, estimate the parameters more accurately, and better generalize to other operating conditions.

CRediT authorship contribution statement

James D. Turner: Conceptualization, Methodology, Software, Validation, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization, Funding acquisition. **Samuel A. Moore:** Writing – original draft, Writing – review & editing. **Brian P. Mann:** Conceptualization, Methodology, Validation, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This project was supported in part by a fellowship award through the National Defense Science and Engineering Graduate (NDSEG) Fellowship Program, sponsored by the Air Force Research Laboratory (AFRL), the Office of Naval Research (ONR), and the Army Research Office (ARO). Additionally, this project was supported in part by U.S. Army Research Office (ARO) grant W911NF-17-0047 and National Science Foundation (NSF) Award 2053470.

Appendix A. Maximum a posteriori estimation as a least squares problem

Consider maximum a posteriori (MAP) estimation of unknown parameters of an ODE or DDE model given noisy measurements of the system's state. Assume that time series of noisy measurements are available, where each measurement $\boldsymbol{z}_j^{[i]}$ in time series i is a known function of the system state $\boldsymbol{x}_j^{[i]}$ with additive Gaussian noise,

$$\boldsymbol{z}_j^{[i]} = \boldsymbol{h}\left(\boldsymbol{x}_j^{[i]}\right) + \boldsymbol{\epsilon}_j^{[i]}, \quad \boldsymbol{\epsilon}_j^{[i]} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}) \tag{A.1}$$

and that all the noise vectors $\boldsymbol{\epsilon}_j^{[i]}$ for all the samples are mutually independent. In other words, assume a multivariate normal density for each measurement,

$$p\left(\boldsymbol{z}_j^{[i]} \mid \boldsymbol{x}_j^{[i]}\right) = \left((2\pi)^{d_z} \det \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}\right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{x}_j^{[i]}\right)\right)^{\top} \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}^{-1}\left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{x}_j^{[i]}\right)\right)\right) \tag{A.2}$$

where $d_{\boldsymbol{z}}$ is the number of elements in each $\boldsymbol{z}_j^{[i]}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$ is the covariance of the noise.

Let $\boldsymbol{\theta}$ be a vector containing all of the unknown model parameters and unknown initial conditions necessary to approximately describe the trajectories of the system corresponding to the time series. Let $\boldsymbol{y}_j^{[i]}$ be the point on the ith approximate trajectory which corresponds to $\boldsymbol{x}_j^{[i]}$. For ODE models, the initial condition at a single instant in time is sufficient for each trajectory, and the trajectories can be obtained numerically or analytically. For DDE models, the spectral element method provides a convenient way to represent the initial segment of each trajectory and compute the approximate trajectories. In summary, each $\boldsymbol{y}_j^{[i]}$ approximating $\boldsymbol{x}_j^{[i]}$ is a known function of $\boldsymbol{\theta}$:

$$\boldsymbol{x}_j^{[i]} \approx \boldsymbol{y}_j^{[i]} = \boldsymbol{f}_j^{[i]}(\boldsymbol{\theta}) \tag{A.3}$$

Assume also that the prior knowledge of $\boldsymbol{\theta}$ can be expressed as a multivariate normal density, $\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\theta}}, \boldsymbol{\Sigma}_{\boldsymbol{\theta}})$, i.e.

$$p(\boldsymbol{\theta}) = \left((2\pi)^{d_\theta} \det \boldsymbol{\Sigma}_{\boldsymbol{\theta}}\right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(\boldsymbol{\theta} - \boldsymbol{\mu}_{\boldsymbol{\theta}}\right)^{\top} \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}\left(\boldsymbol{\theta} - \boldsymbol{\mu}_{\boldsymbol{\theta}}\right)\right) \tag{A.4}$$

where $d_{\boldsymbol{\theta}}$ is the number of elements in $\boldsymbol{\theta}$.

The objective is to find the value of $\boldsymbol{\theta}$ at the maximum posterior density:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}}\ p\left(\boldsymbol{\theta} \mid \boldsymbol{z}_1^{[1]}, \ldots\right) \tag{A.5}$$

Applying Bayes' theorem, this is equivalent to

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}}\ \frac{p\left(\boldsymbol{z}_1^{[1]}, \ldots \mid \boldsymbol{\theta}\right) p(\boldsymbol{\theta})}{\int_{\boldsymbol{\Theta}} p\left(\boldsymbol{z}_1^{[1]}, \ldots \mid \boldsymbol{\theta}'\right) p\left(\boldsymbol{\theta}'\right) \mathrm{d}\boldsymbol{\theta}'} \tag{A.6}$$

which is equivalent to

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}}\ p\left(\boldsymbol{z}_1^{[1]}, \ldots \mid \boldsymbol{\theta}\right) p(\boldsymbol{\theta}) \tag{A.7}$$

since the denominator of Eq. (A.6) is always positive and does not depend on $\boldsymbol{\theta}$. Using Eq. (A.3), since all $\boldsymbol{y}_j^{[i]}$ are known functions of $\boldsymbol{\theta}$, this can be approximately expressed as

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}}\ p\left(\boldsymbol{z}_1^{[1]}, \ldots \mid \boldsymbol{y}_1^{[1]}, \ldots\right) p(\boldsymbol{\theta}) \tag{A.8}$$

This can be simplified as follows. First, since the noise vectors are assumed to be independent, the joint likelihood is equal to the product of the likelihoods of the individual noise vectors,

$$p\left(\boldsymbol{z}_1^{[1]}, \ldots \mid \boldsymbol{y}_1^{[1]}, \ldots\right) = \prod_{i,j} p\left(\boldsymbol{z}_j^{[i]} \mid \boldsymbol{y}_j^{[i]}\right) \tag{A.9}$$

Then, since the posterior density is always nonnegative and the natural logarithm is a monotonically increasing function, maximizing the posterior density is equivalent to maximizing its natural logarithm, which can then be expanded into a sum:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}}\ p\left(\boldsymbol{z}_1^{[1]}, \ldots \mid \boldsymbol{y}_1^{[1]}, \ldots\right) p(\boldsymbol{\theta}) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left(\prod_{i,j} p\left(\boldsymbol{z}_j^{[i]} \mid \boldsymbol{y}_j^{[i]}\right)\right) p(\boldsymbol{\theta}) \tag{A.10}$$

$$= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \ln\left(\left(\prod_{i,j} p\left(\boldsymbol{z}_j^{[i]} \mid \boldsymbol{y}_j^{[i]}\right)\right) p(\boldsymbol{\theta})\right) \tag{A.11}$$

$$= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left(\sum_{i,j} \ln p\left(\boldsymbol{z}_j^{[i]} \mid \boldsymbol{y}_j^{[i]}\right) + \ln p(\boldsymbol{\theta})\right) \tag{A.12}$$

Let

$$\boldsymbol{r}_j^{[i]} = \boldsymbol{L}_\epsilon^\top \left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{y}_j^{[i]}\right)\right) \tag{A.13}$$

where $\boldsymbol{L}_\epsilon$ is a factor in the Cholesky decomposition $\boldsymbol{\Sigma}_\epsilon^{-1} = \boldsymbol{L}_\epsilon \boldsymbol{L}_\epsilon^\top$, so that

$$\left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{x}_j^{[i]}\right)\right)^\top \boldsymbol{\Sigma}_\epsilon^{-1} \left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{x}_j^{[i]}\right)\right) \tag{A.14}$$

$$\approx \left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{y}_j^{[i]}\right)\right)^\top \boldsymbol{\Sigma}_\epsilon^{-1} \left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{y}_j^{[i]}\right)\right) \tag{A.15}$$

$$= \left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{y}_j^{[i]}\right)\right)^\top \boldsymbol{L}_\epsilon \boldsymbol{L}_\epsilon^\top \left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{y}_j^{[i]}\right)\right) \tag{A.16}$$

$$= \left(\boldsymbol{L}_\epsilon^\top \left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{y}_j^{[i]}\right)\right)\right)^\top \boldsymbol{L}_\epsilon^\top \left(\boldsymbol{z}_j^{[i]} - \boldsymbol{h}\left(\boldsymbol{y}_j^{[i]}\right)\right) \tag{A.17}$$

$$= \left(\boldsymbol{r}_j^{[i]}\right)^\top \boldsymbol{r}_j^{[i]} \tag{A.18}$$

Similarly, let

$$\boldsymbol{r}_{\boldsymbol{\theta}} = \boldsymbol{L}_{\boldsymbol{\theta}}^\top \left(\boldsymbol{\theta} - \boldsymbol{\mu}_{\boldsymbol{\theta}}\right) \tag{A.19}$$

where $\boldsymbol{L}_{\boldsymbol{\theta}}$ is a factor in the Cholesky decomposition $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} = \boldsymbol{L}_{\boldsymbol{\theta}} \boldsymbol{L}_{\boldsymbol{\theta}}^\top$, so that

$$\left(\boldsymbol{\theta} - \boldsymbol{\mu}_{\boldsymbol{\theta}}\right)^\top \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \left(\boldsymbol{\theta} - \boldsymbol{\mu}_{\boldsymbol{\theta}}\right) = \left(\boldsymbol{r}_{\boldsymbol{\theta}}\right)^\top \boldsymbol{r}_{\boldsymbol{\theta}} \tag{A.20}$$

Then, substituting Eqs. (A.2), (A.4), (A.18), and (A.20) into Eq. (A.12), the optimization problem becomes

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left(\sum_{i,j} \ln\left(\left((2\pi)^{d_z} \det \boldsymbol{\Sigma}_\epsilon\right)^{-1/2} \exp\left(-\frac{1}{2}\left(\boldsymbol{r}_j^{[i]}\right)^\top \boldsymbol{r}_j^{[i]}\right)\right) + \ln\left(\left((2\pi)^{d_{\boldsymbol{\theta}}} \det \boldsymbol{\Sigma}_{\boldsymbol{\theta}}\right)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(\boldsymbol{r}_{\boldsymbol{\theta}}\right)^\top \boldsymbol{r}_{\boldsymbol{\theta}}\right)\right)\right) \tag{A.21}$$

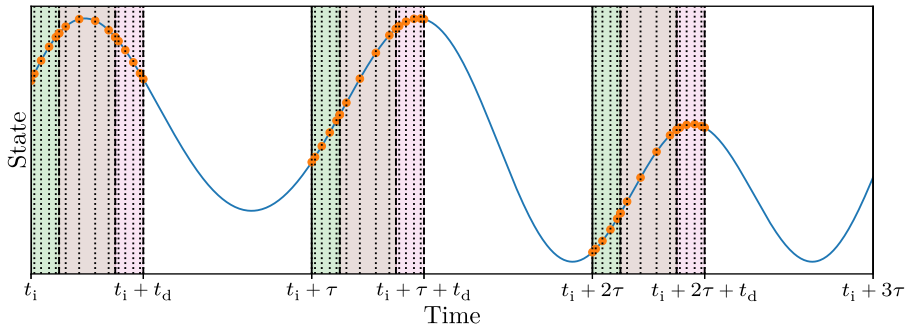


Fig. B.15. Illustration of the spectral element segments, elements, and nodes for a system with piecewise dynamics described by Eqs. (B.1) and (B.2). The state of the system is indicated with a curved blue line. The divisions between segments are indicated with solid vertical lines. The end of each element is indicated with a vertical dashed line. The location of each node is indicated with a vertical line (solid, dashed, or dotted), and the corresponding value of the state is indicated with an orange circular marker. The intervals of time following Eq. (B.1) have a shaded background, while the intervals of time following Eq. (B.2) have a white background. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

$$= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left(\sum_{i,j} \left(\ln\left((2\pi)^{d_z} \det \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \right)^{-1/2} - \frac{1}{2} \left(\boldsymbol{r}_j^{[i]} \right)^\mathsf{T} \boldsymbol{r}_j^{[i]} \right) + \ln\left((2\pi)^{d_\theta} \det \boldsymbol{\Sigma}_{\boldsymbol{\theta}} \right)^{-\frac{1}{2}} - \frac{1}{2} \left(\boldsymbol{r}_{\boldsymbol{\theta}} \right)^\mathsf{T} \boldsymbol{r}_{\boldsymbol{\theta}} \right) \tag{A.22}$$

$$= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left(\sum_{i,j} \ln\left((2\pi)^{d_z} \det \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}} \right)^{-1/2} - \frac{1}{2} \sum_{i,j} \left(\boldsymbol{r}_j^{[i]} \right)^\mathsf{T} \boldsymbol{r}_j^{[i]} + \ln\left((2\pi)^{d_\theta} \det \boldsymbol{\Sigma}_{\boldsymbol{\theta}} \right)^{-\frac{1}{2}} - \frac{1}{2} \left(\boldsymbol{r}_{\boldsymbol{\theta}} \right)^\mathsf{T} \boldsymbol{r}_{\boldsymbol{\theta}} \right) \tag{A.23}$$

Subtracting the terms which do not depend on $\boldsymbol{\theta}$ yields

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left(-\frac{1}{2} \sum_{i,j} \left(\boldsymbol{r}_j^{[i]} \right)^\mathsf{T} \boldsymbol{r}_j^{[i]} - \frac{1}{2} \left(\boldsymbol{r}_{\boldsymbol{\theta}} \right)^\mathsf{T} \boldsymbol{r}_{\boldsymbol{\theta}} \right) \tag{A.24}$$

Dividing by $-1/2$ changes this into an equivalent minimization problem:

$$\underset{\boldsymbol{\theta}}{\operatorname{argmin}} \left(\sum_{i,j} \left(\boldsymbol{r}_j^{[i]} \right)^\mathsf{T} \boldsymbol{r}_j^{[i]} + \left(\boldsymbol{r}_{\boldsymbol{\theta}} \right)^\mathsf{T} \boldsymbol{r}_{\boldsymbol{\theta}} \right) \tag{A.25}$$

This is an unconstrained, nonlinear, least-squares optimization problem. Solving it finds the MAP estimate of $\boldsymbol{\theta}$ according to the prior, measurements, and models.

Appendix B. Adapting the spectral element method

Consider a system with piecewise dynamics,

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(t,\boldsymbol{p})\boldsymbol{x}(t) + \boldsymbol{B}(t,\boldsymbol{p})\boldsymbol{x}(t-\tau) + \boldsymbol{c}(t,\boldsymbol{p}) \qquad \text{if } 0 \le t - t_\mathrm{b}(t) < t_\mathrm{d} \tag{B.1}$$

$$\boldsymbol{x}(t) = \boldsymbol{D}(t - t_\mathrm{s}(t), \boldsymbol{p}) \left(\boldsymbol{x}(t_\mathrm{s}(t)) - \boldsymbol{e}(t_\mathrm{s}(t), \boldsymbol{p}) \right) + \boldsymbol{e}(t,\boldsymbol{p}) \qquad \text{if } t_\mathrm{d} \le t - t_\mathrm{b}(t) < \tau \tag{B.2}$$

where $\boldsymbol{A}(t,\boldsymbol{p})$, $\boldsymbol{B}(t,\boldsymbol{p})$, $\boldsymbol{c}(t,\boldsymbol{p})$, and $\boldsymbol{e}(t,\boldsymbol{p})$ are functions with period T; T is an integer multiple of τ; and

$$t_\mathrm{b}(t) = \tau \lfloor (t - t_\mathrm{i})/\tau \rfloor + t_\mathrm{i} \tag{B.3}$$

$$t_\mathrm{s}(t) = t_\mathrm{b}(t) + t_\mathrm{d} \tag{B.4}$$

Time is split into segments of duration τ, starting at an initial time t_i. Within each segment of duration τ, the system follows Eq. (B.1) for the initial interval of duration t_d, and then it switches to Eq. (B.2) for the remaining interval of duration $\tau - t_\mathrm{d}$. Note that the milling model used in this paper is of this form—Eq. (4) corresponds to Eq. (B.1), and Eq. (34) corresponds to Eq. (B.2)—but the spectral element method described here can also be used for any other system with dynamics of this form.

The solution for a system with dynamics described by Eqs. (B.1) and (B.2) can be approximated using an extended version of the spectral element method presented in [66]. The approach and derivation used here is similar, but is extended to handle nonzero $\boldsymbol{c}(t,\boldsymbol{p})$, piecewise dynamics, and coefficients whose period is an integer multiple of τ.

The approximate solution is represented by the value of the solution at discrete nodes, as illustrated by the circular markers in Fig. B.15. The initial interval of duration t_d in each segment is split into n_e elements with indices $\{0, \ldots, n_\mathrm{e} - 1\}$. Within each element, the solution is approximated by a polynomial of order n_o which interpolates between $n_\mathrm{o} + 1$ nodes with indices $\{0, \ldots, n_\mathrm{o}\}$. Let $t_{i,j,k}$ be the time corresponding to node k in element j in segment i. The times of the nodes within segments are consistent,

i.e. $t_{i,j,k} = t_{i-1,j,k} + \tau$, and consecutive elements within a segment are contiguous, i.e. $t_{i,j,0} = t_{i,j-1,n_o}$ for $j > 0$. For element j in segment i, the solution is approximated by

$$\boldsymbol{x}(t) \approx \sum_{k=0}^{n_o} \boldsymbol{a}_{i,j,k} \phi_k(\nu_{i,j}(t)), \qquad t \in \left[t_{i,j,0}, t_{i,j,n_o}\right] \tag{B.5}$$

where $\boldsymbol{a}_{i,j,k}$ is the solution at time $t_{i,j,k}$; $\nu_{i,j}(t) = (t - t_{i,j,0})/(t_{i,j,n_o} - t_{i,j,0})$ is a function shifting and scaling the time t to a coordinate in $[0,1]$ within the element; and $\phi_k : [0,1] \to \mathbb{R}$ are Lagrange trial functions of order n_o. The nodes are chosen to be the Legendre–Gauss–Lobatto (LGL) points of the specified order within the element, and the trial functions are chosen to be Lagrange polynomials, such that

$$\phi_k(\nu_{i,j}(t_{i,j,m})) = \begin{cases} 1 & k = m \\ 0 & k \neq m \end{cases} \tag{B.6}$$

For times following the initial interval of duration t_d in each segment i, the dynamics follow Eq. (B.2), so the solution in this interval for segment i is approximated by

$$\boldsymbol{x}(t) \approx \boldsymbol{D}(t - t_{i,n_e-1,n_o}, \boldsymbol{p}) \left(\boldsymbol{a}_{i,n_e-1,n_o} - \boldsymbol{e}(t_{i,n_e-1,n_o}, \boldsymbol{p})\right) + \boldsymbol{e}(t, \boldsymbol{p}), \qquad t \in \left[t_{i,n_e-1,n_o}, t_{i+1,0,0}\right] \tag{B.7}$$

Applying this to the full interval of duration $\tau - t_d$ between the end of the last element in previous segment and the start of the current segment yields

$$\boldsymbol{a}_{i,0,0} = \boldsymbol{D}(\tau - t_d, \boldsymbol{p}) \left(\boldsymbol{a}_{i-1,n_e-1,n_o} - \boldsymbol{e}(t_{i-1,n_e-1,n_o}, \boldsymbol{p})\right) + \boldsymbol{e}(t_{i,0,0}, \boldsymbol{p}) \tag{B.8}$$

$$= \boldsymbol{D}(\tau - t_d, \boldsymbol{p}) \boldsymbol{a}_{i-1,n_e-1,n_o} + \tilde{\boldsymbol{d}}_i \tag{B.9}$$

where

$$\tilde{\boldsymbol{d}}_i = \boldsymbol{e}(t_{i,0,0}, \boldsymbol{p}) - \boldsymbol{D}(\tau - t_d, \boldsymbol{p}) \boldsymbol{e}(t_{i-1,n_e-1,n_o}, \boldsymbol{p}) \tag{B.10}$$

Approximate expressions for the time derivative and delayed state within each element can be derived from Eq. (B.5). Note that the value of the derivative $\nu'_{i,j}(t)$ is independent of i and t; this derivative will be denoted ν'_j. Then, for element j in segment i,

$$\dot{\boldsymbol{x}}(t) \approx \sum_{k=0}^{n_o} \boldsymbol{a}_{i,j,k} \nu'_j \phi'_k(\nu_{i,j}(t)), \qquad t \in \left[t_{i,j,0}, t_{i,j,n_o}\right] \tag{B.11}$$

$$\boldsymbol{x}(t - \tau) \approx \sum_{k=0}^{n_o} \boldsymbol{a}_{i-1,j,k} \phi_k(\nu_{i,j}(t)), \qquad t \in \left[t_{i,j,0}, t_{i,j,n_o}\right] \tag{B.12}$$

Substituting into Eq. (B.1) and rearranging yields

$$\left(\sum_{k=0}^{n_o} \boldsymbol{a}_{i,j,k} \nu'_j \phi'_k(\nu_{i,j}(t))\right) - \boldsymbol{A}(t, \boldsymbol{p}) \left(\sum_{k=0}^{n_o} \boldsymbol{a}_{i,j,k} \phi_k(\nu_{i,j}(t))\right) \approx \boldsymbol{B}(t, \boldsymbol{p}) \left(\sum_{k=0}^{n_o} \boldsymbol{a}_{i-1,j,k} \phi_k(\nu_{i,j}(t))\right) + \boldsymbol{c}(t, \boldsymbol{p}), \qquad t \in \left[t_{i,j,0}, t_{i,j,n_o}\right] \tag{B.13}$$

Simplifying, this becomes

$$\sum_{k=0}^{n_o} \left(\boldsymbol{I} \nu'_j \phi'_k(\nu_{i,j}(t)) - \boldsymbol{A}(t, \boldsymbol{p}) \phi_k(\nu_{i,j}(t))\right) \boldsymbol{a}_{i,j,k} \approx \left(\sum_{k=0}^{n_o} \boldsymbol{B}(t, \boldsymbol{p}) \phi_k(\nu_{i,j}(t)) \boldsymbol{a}_{i-1,j,k}\right) + \boldsymbol{c}(t, \boldsymbol{p}), \qquad t \in \left[t_{i,j,0}, t_{i,j,n_o}\right] \tag{B.14}$$

The method of weighted residuals can be applied to obtain multiple equations which enforce approximate equality of Eq. (B.14). As in [66], the first n_o Legendre polynomials are used as the test functions. They are denoted ψ_ℓ and have indices $\ell \in \{0, \ldots, n_o - 1\}$. For each element j in segment i:

$$\begin{aligned} &\int_0^1 \left(\sum_{k=0}^{n_o} \left(\boldsymbol{I} \nu'_j \phi'_k(\sigma) - \boldsymbol{A}\left(\nu_{i,j}^{-1}(\sigma), \boldsymbol{p}\right) \phi_k(\sigma)\right) \boldsymbol{a}_{i,j,k}\right) \psi_\ell(\sigma) \, \mathrm{d}\sigma \\ &= \int_0^1 \left(\left(\sum_{k=0}^{n_o} \boldsymbol{B}\left(\nu_{i,j}^{-1}(\sigma), \boldsymbol{p}\right) \phi_k(\sigma) \boldsymbol{a}_{i-1,j,k}\right) + \boldsymbol{c}\left(\nu_{i,j}^{-1}(\sigma), \boldsymbol{p}\right)\right) \psi_\ell(\sigma) \, \mathrm{d}\sigma \end{aligned} \tag{B.15}$$

where $\nu_{i,j}^{-1}(\sigma) = t_{i,j,0} + \sigma \cdot (t_{i,j,n_o} - t_{i,j,0})$ denotes the inverse function of $\nu_{i,j}$. Rearranging, this becomes

$$\begin{aligned} &\sum_{k=0}^{n_o} \left(\int_0^1 \left(\boldsymbol{I} \nu'_j \phi'_k(\sigma) - \boldsymbol{A}\left(\nu_{i,j}^{-1}(\sigma), \boldsymbol{p}\right) \phi_k(\sigma)\right) \psi_\ell(\sigma) \, \mathrm{d}\sigma\right) \boldsymbol{a}_{i,j,k} \\ &= \sum_{k=0}^{n_o} \left(\int_0^1 \boldsymbol{B}\left(\nu_{i,j}^{-1}(\sigma), \boldsymbol{p}\right) \phi_k(\sigma) \psi_\ell(\sigma) \, \mathrm{d}\sigma\right) \boldsymbol{a}_{i-1,j,k} + \int_0^1 \boldsymbol{c}\left(\nu_{i,j}^{-1}(\sigma), \boldsymbol{p}\right) \psi_\ell(\sigma) \, \mathrm{d}\sigma \end{aligned} \tag{B.16}$$

which can be written as

$$\sum_{k=0}^{n_o} \boldsymbol{N}_{i,j,k,\ell} \boldsymbol{a}_{i,j,k} = \sum_{k=0}^{n_o} \boldsymbol{P}_{i,j,k,\ell} \boldsymbol{a}_{i-1,j,k} + \boldsymbol{d}_{i,j,\ell} \tag{B.17}$$

where

$$\boldsymbol{N}_{i,j,k,\ell} = \int_0^1 \left(\boldsymbol{I} \nu_j' \phi_k'(\sigma) - \boldsymbol{A}(\nu_{i,j}^{-1}(\sigma), \boldsymbol{p}) \phi_k(\sigma) \right) \psi_\ell(\sigma) \, \mathrm{d}\sigma \tag{B.18}$$

$$\boldsymbol{P}_{i,j,k,\ell} = \int_0^1 \boldsymbol{B}(\nu_{i,j}^{-1}(\sigma), \boldsymbol{p}) \phi_k(\sigma) \psi_\ell(\sigma) \, \mathrm{d}\sigma \tag{B.19}$$

$$\boldsymbol{d}_{i,j,\ell} = \int_0^1 \boldsymbol{c}(\nu_{i,j}^{-1}(\sigma), \boldsymbol{p}) \psi_\ell(\sigma) \, \mathrm{d}\sigma \tag{B.20}$$

The integrals can be approximated by LGL quadrature using LGL weights w_m, which is convenient because the nodes were chosen to be LGL points:

$$\boldsymbol{N}_{i,j,k,\ell} \approx \sum_{m=1}^{n_\mathrm{o}} \left(\boldsymbol{I} \nu_j' \phi_k'(\sigma_m) - \boldsymbol{A}\left(t_{i,j,m}, \boldsymbol{p}\right) \phi_k(\sigma_m) \right) \psi_\ell(\sigma_m) w_m \tag{B.21}$$

$$\boldsymbol{P}_{i,j,k,\ell} \approx \sum_{m=1}^{n_\mathrm{o}} \boldsymbol{B}\left(t_{i,j,m}, \boldsymbol{p}\right) \phi_k(\sigma_m) \psi_\ell(\sigma_m) w_m \tag{B.22}$$

$$\boldsymbol{d}_{i,j,\ell} \approx \sum_{m=1}^{n_\mathrm{o}} \boldsymbol{c}\left(t_{i,j,m}, \boldsymbol{p}\right) \psi_\ell(\sigma_m) w_m \tag{B.23}$$

where $\sigma_m = \nu_{i,j}(t_{i,j,m})$. Substituting Eq. (B.6), this simplifies to

$$\boldsymbol{N}_{i,j,k,\ell} \approx \left(\sum_{m=1}^{n_\mathrm{o}} \nu_j' \phi_k'(\sigma_m) \psi_\ell(\sigma_m) w_m \right) \boldsymbol{I} - \boldsymbol{A}\left(t_{i,j,k}, \boldsymbol{p}\right) \phi_k(\sigma_k) \psi_\ell(\sigma_k) w_k \tag{B.24}$$

$$\boldsymbol{P}_{i,j,k,\ell} \approx \boldsymbol{B}\left(t_{i,j,k}, \boldsymbol{p}\right) \phi_k(\sigma_k) \psi_\ell(\sigma_k) w_k \tag{B.25}$$

$$\boldsymbol{d}_{i,j,\ell} \approx \sum_{m=1}^{n_\mathrm{o}} \boldsymbol{c}(t_{i,j,m}, \boldsymbol{p}) \psi_\ell(\sigma_m) w_m \tag{B.26}$$

Combining Eq. (B.8) with Eq. (B.17) for all the elements and trial functions yields the following relationship between the solution in segment i and the solution in segment $i-1$:

$$\bar{\boldsymbol{N}}_i \bar{\boldsymbol{a}}_i = \bar{\boldsymbol{P}}_i \bar{\boldsymbol{a}}_{i-1} + \bar{\boldsymbol{d}}_i \tag{B.27}$$

For example, for $n_\mathrm{e} = 2$,

$$\bar{\boldsymbol{N}}_i = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{N}_{i,0,0,0} & \boldsymbol{N}_{i,0,1,0} & \cdots & \boldsymbol{N}_{i,0,n_\mathrm{o},0} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{N}_{i,0,0,n_\mathrm{o}-1} & \boldsymbol{N}_{i,0,1,n_\mathrm{o}-1} & \cdots & \boldsymbol{N}_{i,0,n_\mathrm{o},n_\mathrm{o}-1} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{N}_{i,1,0,0} & \boldsymbol{N}_{i,1,1,0} & \cdots & \boldsymbol{N}_{i,1,n_\mathrm{o},0} \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \vdots & \ddots & \vdots & \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{N}_{i,1,0,n_\mathrm{o}-1} & \boldsymbol{N}_{i,1,1,n_\mathrm{o}-1} & \cdots & \boldsymbol{N}_{i,1,n_\mathrm{o},n_\mathrm{o}-1} \end{bmatrix} \tag{B.28}$$

$$\bar{\boldsymbol{P}}_i = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{D}(\tau - t_\mathrm{d}, \boldsymbol{p}) \\ \boldsymbol{P}_{i,0,0,0} & \boldsymbol{P}_{i,0,1,0} & \cdots & \boldsymbol{P}_{i,0,n_\mathrm{o},0} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{P}_{i,0,0,n_\mathrm{o}-1} & \boldsymbol{P}_{i,0,1,n_\mathrm{o}-1} & \cdots & \boldsymbol{P}_{i,0,n_\mathrm{o},n_\mathrm{o}-1} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{P}_{i,1,0,0} & \boldsymbol{P}_{i,1,1,0} & \cdots & \boldsymbol{P}_{i,1,n_\mathrm{o},0} \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \vdots & \ddots & \vdots & \\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{P}_{i,1,0,n_\mathrm{o}-1} & \boldsymbol{P}_{i,1,1,n_\mathrm{o}-1} & \cdots & \boldsymbol{P}_{i,1,n_\mathrm{o},n_\mathrm{o}-1} \end{bmatrix} \tag{B.29}$$

$$\bar{\boldsymbol{a}}_i = \begin{bmatrix} \boldsymbol{a}_{i,0,0} \\ \boldsymbol{a}_{i,0,1} \\ \vdots \\ \boldsymbol{a}_{i,0,n_\mathrm{o}} = \boldsymbol{a}_{i,1,0} \\ \boldsymbol{a}_{i,1,1} \\ \vdots \\ \boldsymbol{a}_{i,1,n_\mathrm{o}} \end{bmatrix}, \qquad \bar{\boldsymbol{a}}_{i-1} = \begin{bmatrix} \boldsymbol{a}_{i-1,0,0} \\ \boldsymbol{a}_{i-1,0,1} \\ \vdots \\ \boldsymbol{a}_{i-1,0,n_\mathrm{o}} = \boldsymbol{a}_{i-1,1,0} \\ \boldsymbol{a}_{i-1,1,1} \\ \vdots \\ \boldsymbol{a}_{i-1,1,n_\mathrm{o}} \end{bmatrix}, \qquad \bar{\boldsymbol{d}}_i = \begin{bmatrix} \tilde{\boldsymbol{d}}_i \\ \boldsymbol{d}_{i,0,0} \\ \vdots \\ \boldsymbol{d}_{i,0,n_\mathrm{o}-1} \\ \boldsymbol{d}_{i,1,0} \\ \vdots \\ \boldsymbol{d}_{i,1,n_\mathrm{o}-1} \end{bmatrix} \tag{B.30}$$

Eq. (B.27) can be solved for $\bar{a}_i$ to obtain

$$\bar{\boldsymbol{a}}_i = \bar{\boldsymbol{Q}}_i \bar{\boldsymbol{a}}_{i-1} + \bar{\boldsymbol{r}}_i \qquad \text{where} \qquad \bar{\boldsymbol{Q}}_i = \bar{\boldsymbol{N}}_i^{-1} \bar{\boldsymbol{P}}_i, \quad \bar{\boldsymbol{r}}_i = \bar{\boldsymbol{N}}_i^{-1} \bar{\boldsymbol{d}}_i \tag{B.31}$$

Observe that $\bar{\boldsymbol{Q}}_i = \bar{\boldsymbol{Q}}_{i+T/\tau}$ and $\bar{\boldsymbol{r}}_i = \bar{\boldsymbol{r}}_{i+T/\tau}$ due to the periodicity of $\boldsymbol{A}(t,\boldsymbol{p})$, $\boldsymbol{B}(t,\boldsymbol{p})$, $\boldsymbol{c}(t,\boldsymbol{p})$, and $\boldsymbol{e}(t,\boldsymbol{p})$. As a result, $\bar{\boldsymbol{Q}}_1,\dots,\bar{\boldsymbol{Q}}_{T/\tau}$ and $\bar{\boldsymbol{r}}_1,\dots,\bar{\boldsymbol{r}}_{T/\tau}$ are sufficient to approximately describe the dynamics of the system. For example, given $\bar{\boldsymbol{a}}_0$, they can be used to compute $\bar{\boldsymbol{a}}_1,\bar{\boldsymbol{a}}_2,\dots$, which can be interpolated using Eqs. (B.5) and (B.7) to find the approximate solution at an arbitrary future time.

For computation of the CMs, the equations described by Eq. (B.31) for all $i = 1,\dots,T/\tau$ can be combined into a single equation describing the mapping over an entire period T,

$$\bar{\boldsymbol{a}}_{(i+1)\cdot(T/\tau)} = \check{\boldsymbol{Q}}_{1,\dots,T/\tau}\bar{\boldsymbol{a}}_{i\cdot(T/\tau)} + \check{\boldsymbol{r}}_{1,\dots,T/\tau} \tag{B.32}$$

where $\check{\boldsymbol{Q}}_{1,\dots,T/\tau} = \bar{\boldsymbol{Q}}_{T/\tau}\cdots\bar{\boldsymbol{Q}}_1$ is a constant matrix product and $\check{\boldsymbol{r}}_{1,\dots,T/\tau}$ is a constant vector. The equilibrium point is given by

$$\bar{\boldsymbol{a}}^* = \left(\boldsymbol{I} - \check{\boldsymbol{Q}}_{1,\dots,T/\tau}\right)^{-1}\check{\boldsymbol{r}}_{1,\dots,T/\tau} \tag{B.33}$$

which can be substituted to obtain

$$\bar{\boldsymbol{a}}_{(i+1)\cdot(T/\tau)} - \bar{\boldsymbol{a}}^* = \check{\boldsymbol{Q}}_{1,\dots,T/\tau}\left(\bar{\boldsymbol{a}}_{i\cdot(T/\tau)} - \bar{\boldsymbol{a}}^*\right) \tag{B.34}$$

So, the system converges, i.e. $\lim_{i\to\infty}\bar{\boldsymbol{a}}_{i\cdot(T/\tau)} = \bar{\boldsymbol{a}}^*$, if the all eigenvalues of $\check{\boldsymbol{Q}}_{1,\dots,T/\tau}$ have magnitude less than 1, and diverges otherwise. In other words, the eigenvalues of $\check{\boldsymbol{Q}}_{1,\dots,T/\tau}$ are the CMs of the spectral element approximation. They approximately describe the stability of the system being approximated.

References

[1] T. Insperger, G. Stépán, P.V. Bayly, B.P. Mann, Multiple chatter frequencies in milling processes, J. Sound Vib. 262 (2) (2003) 333–345, http://dx.doi.org/10.1016/S0022-460X(02)01131-8.

[2] B.P. Mann, K.A. Young, T.L. Schmitz, D.N. Dilley, Simultaneous stability and surface location error predictions in milling, J. Manuf. Sci. Eng. 127 (3) (2005) 446–453, http://dx.doi.org/10.1115/1.1948394.

[3] B.P. Mann, B.T. Edes, S.J. Easley, K.A. Young, K. Ma, Chatter vibration and surface location error prediction for helical end mills, Int. J. Mach. Tools Manuf. 48 (3) (2008) 350–361, http://dx.doi.org/10.1016/j.ijmachtools.2007.10.003.

[4] D.J. Tweten, G.M. Lipp, F.A. Khasawneh, B.P. Mann, On the comparison of semi-analytical methods for the stability analysis of delay differential equations, J. Sound Vib. 331 (17) (2012) 4057–4071, http://dx.doi.org/10.1016/j.jsv.2012.04.009.

[5] F.A. Khasawneh, B.P. Mann, T. Insperger, G. Stépán, Increased stability of low-speed turning through a distributed force and continuous delay model, J. Comput. Nonlinear Dynam. 4 (4) (2009) http://dx.doi.org/10.1115/1.3187153.

[6] Y. Altintas, E. Budak, Analytical prediction of stability lobes in milling, CIRP Ann. 44 (1) (1995) 357–362.

[7] T. Insperger, B.P. Mann, G. Stépán, P.V. Bayly, Stability of up-milling and down-milling, part 1: Alternative analytical methods, Int. J. Mach. Tools Manuf. 43 (1) (2003) 25–34, http://dx.doi.org/10.1016/S0890-6955(02)00159-1.

[8] O.A. Bobrenkov, F.A. Khasawneh, E.A. Butcher, B.P. Mann, Analysis of milling dynamics for simultaneously engaged cutting teeth, J. Sound Vib. 329 (5) (2010) 585–606, http://dx.doi.org/10.1016/j.jsv.2009.09.032.

[9] J. Tlusty, Dynamics of high-speed milling, J. Eng. Ind. 108 (1986) 59–67.

[10] T.L. Schmitz, D.G. Scott, Three-component receptance coupling substructure analysis for tool point dynamics prediction, J. Manuf. Sci. Eng. 127 (2005) 781–790.

[11] G. Yücesan, Y. Altintaş, Improved modelling of cutting force coefficients in peripheral milling, Int. J. Mach. Tools Manuf. 34 (4) (1994) 473–487, http://dx.doi.org/10.1016/0890-6955(94)90079-5.

[12] Y. Altintas, Manufacturing Automation: Metal Cutting Mechanics, Machine Tool Vibrations, and CNC Design, second ed., Cambridge University Press, 2012.

[13] B.P. Mann, T. Insperger, P.V. Bayly, G. Stépán, Stability of up-milling and down-milling, part 2: Experimental verification, Int. J. Mach. Tools Manuf. 43 (1) (2003) 35–40, http://dx.doi.org/10.1016/S0890-6955(02)00160-8.

[14] B.P. Mann, Dynamic Models of Milling and Broaching (Ph.D. thesis), Washington University, Sever Institute of Technology, Department of Mechanical Engineering, Saint Louis, Missouri, 2003.

[15] S.A. Tobias, Machine Tool Vibration, Blackie, London, 1965.

[16] J. Tlusty, A. Polacek, C. Danek, J. Spacek, Selbsterregte Schwingungen an Werkzeugmaschinen, VEB Verlag Technik, Berlin, 1962.

[17] H. Merritt, Theory of self-excited machine tool chatter, J. Eng. Ind. 87 (4) (1965) 447–454, http://dx.doi.org/10.1115/1.3670861.

[18] J.R. Pratt, A.H. Nayfeh, Design and modeling for chatter control, Nonlinear Dynam. 19 (1) (1999) 49–69, http://dx.doi.org/10.1023/A:1008322520352.

[19] F. Koenisberger, J. Tlusty, Stability against chatter, in: Machine Tool Structures, Vol. 1, Pergamon Press, 1970, pp. 115–344, http://dx.doi.org/10.1016/C2013-0-02227-X.

[20] R.L. Kegg, Cutting dynamics in machine tool chatter, J. Eng. Ind. 87 (4) (1965) 464–470, http://dx.doi.org/10.1115/1.3670863.

[21] R. Shridar, R.E. Hohn, G.W. Long, A stability algorithm for the general milling process, J. Eng. Ind. 90 (2) (1968) 330–334, http://dx.doi.org/10.1115/1.3604637.

[22] N.H. Hanna, S.A. Tobias, A theory of nonlinear regenerative chatter, J. Eng. Ind. 96 (1) (1974) 247–255, http://dx.doi.org/10.1115/1.3438305.

[23] J. Tlusty, F. Ismail, Special aspects of chatter in milling, J. Vib. Acoust. Stress Reliab. Des. 105 (1) (1983) 24–32, http://dx.doi.org/10.1115/1.3269061.

[24] I. Grabec, Chaotic dynamics of the cutting process, Int. J. Mach. Tools Manuf. 28 (1) (1988) 19–32, http://dx.doi.org/10.1016/0890-6955(88)90004-1.

[25] G. Stépán, Retarded Dynamical Sytems: Stability and Characteristic Functions, John Wiley & Sons, 1989.

[26] I. Minis, R. Yanushevsky, A new theoretical approach for the prediction of machine tool chatter in milling, J. Eng. Ind. 115 (1) (1993) 1–8, http://dx.doi.org/10.1115/1.2901633.

[27] S. Smith, J. Tlusty, An overview of modeling and simulation of the milling process, J. Eng. Ind. 113 (2) (1991) 169–175, http://dx.doi.org/10.1115/1.2899674.

[28] Y. Altintaş, E. Budak, Analytical prediction of stability lobes in milling, CIRP Ann. 44 (1) (1995) 357–362, http://dx.doi.org/10.1016/S0007-8506(07)62342-7.

[29] H. Schulz, T. Moriwaki, High-speed machining, CIRP Ann. 41 (2) (1992) 637–643, http://dx.doi.org/10.1016/S0007-8506(07)63250-8.

[30] A.H. Nayfeh, C.M. Chin, J. Pratt, Applications of perturbation methods to tool chatter dynamics, in: F.C. Moon (Ed.), Dynamics and Chaos in Manufacturing Processes, John Wiley & Sons, Inc., New York, NY, 1997, pp. 193–214.

[31] B. Balachandran, Nonlinear dynamics of milling processes, Philos. Trans. R. Soc. A 359 (1781) (2001) 793–819, http://dx.doi.org/10.1098/rsta.2000.0755.

[32] M.A. Davies, B. Balachandran, Impact dynamics in milling of thin-walled structures, Nonlinear Dynam. 22 (2000) 375–392, http://dx.doi.org/10.1023/A:1008364405411.

[33] Y. Altintas, Analytical prediction of three dimensional chatter stability in milling, JSME Int. J. 44 (3) (2001) 717–723, http://dx.doi.org/10.1299/jsmec.44.717.

[34] R.G. Landers, A.G. Ulsoy, Nonlinear feed effect in machining chatter analysis, J. Manuf. Sci. Eng. 130 (1) (2008) http://dx.doi.org/10.1115/1.2783276.

[35] D. Montgomery, Y. Altintas, Mechanism of cutting force and surface generation in dynamic milling, J. Eng. Ind. 113 (2) (1991) 160–168, http://dx.doi.org/10.1115/1.2899673.

[36] W.T. Corpus, W.J. Endres, A high order solution for the added stability lobes in intermittent machining, in: Proceedings of the ASME Symposium on Machining Processes, 2000, pp. 871–878.

[37] M.A. Davies, J.R. Pratt, B. Dutterer, T.J. Burns, Stability prediction for low radial immersion milling, J. Manuf. Sci. Eng. 124 (2) (2002) 217–225, http://dx.doi.org/10.1115/1.1455030.

[38] X.-H. Long, B. Balachandran, B.P. Mann, Dynamics of milling processes with variable time delays, Nonlinear Dynam. 47 (4) (2007) 49–63, http://dx.doi.org/10.1007/s11071-006-9058-4.

[39] T.L. Schmitz, B.P. Mann, Closed-form solutions for surface location error in milling, Int. J. Mach. Tools Manuf. 46 (12) (2006) 1369–1377, http://dx.doi.org/10.1016/j.ijmachtools.2005.10.007.

[40] B.P. Mann, K.A. Young, An empirical approach for delayed oscillator stability and parametric identification, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 462 (2071) (2006) 2145–2160, http://dx.doi.org/10.1098/rspa.2006.1677.

[41] B.P. Mann, P.V. Bayly, M.A. Davies, J.E. Halley, Limit cycles, bifurcations, and accuracy of the milling process, J. Sound Vib. 277 (1) (2004) 31–48, http://dx.doi.org/10.1016/j.jsv.2003.08.040.

[42] G. Stépán, Delay-differential equation models for machine tool chatter, in: F.C. Moon (Ed.), Dynamics and Chaos in Manufacturing Processes, John Wiley & Sons, Inc., New York, NY, 1997, pp. 165–192.

[43] B.R. Patel, B.P. Mann, K.A. Young, Uncharted islands of chatter instability in milling, Int. J. Mach. Tools Manuf. 48 (1) (2008) 124–134, http://dx.doi.org/10.1016/j.ijmachtools.2007.06.009.

[44] E. Butcher, B. Mann, Stability analysis and control of linear periodic delayed systems using Chebyshev and temporal finite element methods, in: D.E. Gilsinn, T. Kalmár-Nagy, B. Balachandran (Eds.), Delay Differential Equations: Recent Advances and New Directions, Springer, Boston, MA, 2009, pp. 93–129, http://dx.doi.org/10.1007/978-0-387-85595-0_4.

[45] B.P. Mann, B.R. Patel, Stability of delay equations written as state space models, J. Vib. Control 16 (7–8) (2010) 1067–1085, http://dx.doi.org/10.1177/1077546309341111.

[46] E.A. Butcher, O.A. Bobrenkov, On the Chebyshev spectral continuous time approximation for constant and periodic delay differential equations, Commun. Nonlinear Sci. Numer. Simul. 16 (3) (2011) 1541–1554, http://dx.doi.org/10.1016/j.cnsns.2010.05.037.

[47] A.G. Ulsoy, R. Gitik, On the convergence of the matrix Lambert W approach to solution of systems of delay differential equations, J. Dyn. Syst. Meas. Control 142 (2) (2019) http://dx.doi.org/10.1115/1.4045368.

[48] R.G. Landers, G. Galecki, K. Young, R. Hanks, Peripheral milling of thin titanium plates: modelling, analysis, and process planning, Proc. Inst. Mech. Eng. B 225 (6) (2011) 783–798, http://dx.doi.org/10.1177/09544054JEM1994.

[49] M.H. Kurdi, R.T. Haftka, T.L. Schmitz, B.P. Mann, A robust semi-analytical method for calculating the response sensitivity of a time delay system, J. Vib. Acoust. 130 (6) (2008) http://dx.doi.org/10.1115/1.2981093.

[50] G. Urbikain, D. Olvera, L.N. López de Lacalle, A. Elías-Zúñiga, Spindle speed variation technique in turning operations: Modeling and real implementation, J. Sound Vib. 383 (2016) 384–396.

[51] A. Yilmaz, E. Al-Regib, J. Ni, Machine tool chatter suppression by multi-level random spindle speed variation, J. Manuf. Sci. Eng. 124 (2) (2002) 208–216.

[52] Y.S. Tarng, J.Y. Kao, E.C. Lee, Chatter suppression in turning operations with a tuned vibration absorber, J. Mater Process. Technol. 105 (1–2) (2000) 55–60.

[53] N.D. Sims, Vibration absorbers for chatter suppression: A new analytical tuning methodology, J. Sound Vib. 301 (3–5) (2007) 592–607.

[54] Y. Yang, W. Dai, Q. Liu, Design and implementation of two-degree-of-freedom tuned mass damper in milling vibration mitigation, J. Sound Vib. 335 (2015) 78–88.

[55] Z. Yao, D. Mei, Z. Chen, Chatter suppression by parametric excitation: Model and experiments, J. Sound Vib. 330 (13) (2011) 2995–3005.

[56] M.C. Yesilli, F.A. Khasawneh, B.P. Mann, Transfer learning for autonomous chatter detection in milling, J. Manuf. Process. 80 (2022) 1–27.

[57] T.L. Schmitz, Chatter recognition by a statistical evaluation of the synchronously sampled audio signal, J. Sound Vib. 262 (3) (2003) 721–730.

[58] S. Smith, T. Delio, Sensor-based chatter detection and avoidance by spindle speed selection, J. Dyn. Syst. Meas. Control 114 (1992) 486–492.

[59] E. Kuljanic, M. Sortino, G. Totis, Multisensor approaches for chatter detection in milling, J. Sound Vib. 312 (4–5) (2008) 672–693.

[60] M.C. Yesilli, F.A. Khasawneh, A. Otto, On transfer learning for chatter detection in turning using wavelet packet transform and ensemble empirical mode decomposition, CIRP J. Manuf. Sci. Technol. 28 (2020) 118–135.

[61] M.C. Yesilli, F.A. Khasawneh, A. Otto, Chatter detection in turning using machine learning and similarity measures of time series via dynamic time warping, J. Manuf. Process. 77 (2022) 190–206.

[62] F.A. Khasawneh, E. Munch, Chatter detection in turning using persistent homology, Mech. Syst. Signal Process. 70 (2016) 527–541.

[63] M. Eynian, In-process identification of modal parameters using dimensionless relationships in milling chatter, Int. J. Mach. Tools Manuf. 143 (2019) 49–62, http://dx.doi.org/10.1016/j.ijmachtools.2019.04.003.

[64] Y.-Y. Ren, M. Wan, W.-H. Zhang, Y. Yang, A review on methods for obtaining dynamical property parameters of machining processes, Mech. Syst. Signal Process. 194 (2023) 110280, http://dx.doi.org/10.1016/j.ymssp.2023.110280.

[65] Y.-P. Liu, Y. Altintas, In-process identification of machine tool dynamics, CIRP J. Manuf. Sci. Technol. 32 (2021) 322–337, http://dx.doi.org/10.1016/j.cirpj.2021.01.007.

[66] F.A. Khasawneh, B.P. Mann, A spectral element approach for the stability of delay systems, Internat. J. Numer. Methods Engrg. 87 (6) (2011) 566–592, http://dx.doi.org/10.1002/nme.3122.

[67] P. Virtanen, R. Gommers, T.E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovskini, P. Peterson, W. Weckesser, J. Bright, S.J. van der Walt, M. Brett, J. Wilson, K.J. Millman, N. Mayorov, A.R.J. Nelson, E. Jones, R. Kern, E. Larson, C.J. Carey, İ. Polat, Y. Feng, E.W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E.A. Quintero, C.R. Harris, A.M. Archibald, A.H. Ribeiro, F. Pedregosa, P. van Mulbregt, SciPy 1.0 Contributors, SciPy 1.0: Fundamental algorithms for scientific computing in Python, Nature Methods 17 (2020) 261–272, http://dx.doi.org/10.1038/s41592-019-0686-2.

[68] B.P. Mann, B.T. Edes, S.J. Easley, K.A. Young, K. Ma, Surface location error and chatter prediction for helical end mills, Int. J. Mach. Tools Manuf. 48 (2008) 350–361.