



BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
DEPARTMENT OF APPLIED MECHANICS

Numerical methods for the stability and stabilizability analysis of delayed dynamical systems

Author:
Dávid LEHOTZKY

Supervisor:
Tamás INSPERGER, DSc

A thesis submitted in the partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

Budapest, 2016

"The very success of science in showing us how deeply ordered the natural world is provides strong grounds for believing that there is an even deeper cause for that order."

John C. Lennox

Acknowledgements

In the first place, I would like to express my deepest gratitude to Dr. Tamás Insperger for his persistent scientific support and for his guidance during my research work.

I would also like to say thanks to all my colleagues at the Department of Applied Mechanics for the great scientific and social community which I could be part of.

I am thankful to Prof. János Turi who hosted me at the University of Texas at Dallas in 2013 during my MSc studies and who has given me many useful professional advice. I am grateful to Dr. Jokin Muñoa who hosted me for a month in Basque Country and gave me the opportunity to visit IK4-IDEKO in 2014. I am also thankful to Prof. Gábor Stépán for the inspiration I have received from him throughout my studies and my academic work.

Finally, I would like to say thanks to my family for being patient and supportive throughout my studies.

I gratefully acknowledge the support received from the Pro Progressio Foundation in the framework of the Pro Progressio PhD Scholarship Program.

This work has been supported by the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Advanced Grant Agreement N°340889, the Hungarian National Science Foundation under grant OTKA-K105433 and the MTA-BME Lendület Human Balancing Research Group N°2016-6/2016. These supports are also gratefully acknowledged.

Contents

Acknowledgements	v
1 Introduction	1
1.1 Outline of the thesis	2
1.2 Mathematical preliminaries	3
1.2.1 Stability of linear DDEs	4
1.2.2 Equivalent forms of delayed systems	5
2 Numerical methods for stability analysis	7
2.1 Pseudospectral tau method	8
2.1.1 Lagrange interpolation	9
2.1.2 Tau approximation	10
2.1.3 Numerical integration	12
2.1.4 Example	12
2.1.5 Selection of node and test function sets	13
2.1.6 Stability of time-periodic systems	13
2.2 Spectral element method	15
2.3 Comparison of spectral methods	23
2.3.1 Exact stability boundaries	24
2.3.2 Exact value of the rightmost root	25
2.3.3 Methods under comparison	25
2.3.4 Results of comparison	27
2.4 Extension to hybrid systems	29
2.4.1 PsT method	34
2.4.2 SE method	36
2.5 New results	43
3 Applications to machine tool chatter	47
3.1 Milling operations	49
3.1.1 Milling models	49
3.1.2 Application of the SE method	51
3.1.3 Extension of the SE method	53
3.1.4 Comparison with other methods	56
3.2 Digital position control in machining	60
3.2.1 Milling process with active damper	61
3.2.2 Milling with controlled workpiece holder	65
3.2.3 Milling with controlled tool holder	67
3.3 New results	70
4 Stabilizability of delayed systems	73
4.1 Stabilizability of turning processes subjected to active damping	73
4.1.1 Modeling	74
4.1.2 Results	75
4.2 The modeling of human balancing with PIDA control	79

4.2.1	Balancing models	80
4.2.2	Stabilizability analysis	83
4.2.3	Results	86
4.3	New results	89
A	Results from numerical analysis	91
A.1	Legendre polynomials	91
A.2	Lobatto-type Legendre–Gauss quadrature	91
A.3	Chebyshev points and polynomials	92
	Bibliography	95

Dedicated to my Grandpa

Chapter 1

Introduction

In the recent decades an increasing number of research papers and books have dealt with time-delay systems in applied sciences. Time delay emerged first in population dynamics but since then its significance has been discovered in many engineering and biological applications. For instance, control systems always involve feedback delays due to finite-time information transmission, signal processing and actuation [24, 71]. Similarly, the nervous system of humans is subjected to delay [27, 62, 80], which affects balancing abilities and may cause movement disorders. Reflex delay of the human nervous system is the main reason for the development of stop-and-go traffic jams [67, 68]. Time delay plays an important role in contact problems such as the "shimmy motion" of wheels [81]. Machine tool vibrations are also explained by the so-called regenerative delay (for details, see Chapter 4.4 in [75] or [2, 78]).

In the above examples, time delay typically has a destabilizing effect, which is manifested in unwanted vibrations or oscillations around the desired steady-state motion. The local stability analysis of these time-delay systems provides the primary characteristics of their behavior around stationary states. As a result, several analytical and numerical methods have been developed for local stability analysis. For instance, the D-subdivision method [75] gives a closed form solution for the stability boundaries of autonomous delay-differential equations in the space of system parameters. While closed form solutions can be derived for the stability boundaries of some autonomous systems, the stability analysis of time-periodic systems usually requires numerical approximation techniques, especially in the presence of time-delays. The related literature provides several numerical methods for the stability analysis of time-periodic time-delay systems, such as the semi-discretization method [29], full discretization method [18], spectral element method [36] or the pseudospectral collocation method [12, 13].

The optimal selection of system parameters plays an important role during the process of design of engineering applications. Stability diagrams show the stable and unstable domains in the space of system parameters of locally linearized systems. For the calculation of stability diagrams the stability of the linearized system has to be determined for multiple sets of system parameters. As a result, the computational properties of the different numerical methods for stability analysis and the development of new, computationally more efficient methods are still important in engineering. Note that the current trend of automation and data exchange in manufacturing, called Industry 4.0, aims to create smart factories, where the physical processes are monitored and virtual copies are created of the real world. Based on the collected information, the different modules of these smart factories make decisions on their own, thus the manufacturing processes of Industry 4.0 involve digital control loops. Consequently, in machining, the analysis of physical processes subjected to digitally controlled systems receives greater attention.

The modeling of such processes leads to hybrid time-delay systems whose analysis is difficult and not well-established by the engineering literature. Note that, generally speaking, delayed processes subjected to digital feedback control are all governed by hybrid time-delayed systems, e.g. the human-machine interactions of haptic systems [40]. This thesis describes two efficient numerical methods which can be used for the stability analysis of hybrid time-delayed systems. Furthermore, the application of these numerical methods is also presented to digitally controlled machining operations.

In addition to the stability analysis of time-delay systems, the condition of stabilizability is often required in many applications. Stabilizability properties describe whether the locally linearized dynamical system can be made stable by the proper choice of control or system parameters. Stabilizability plays an important role in parameter optimization, where, besides keeping the stability of the system, a cost function has to be minimized. This thesis deals with such a problem where the material removal rate of the machining process is maximized in turning processes subjected to active damping by the proper selection of the control gains of the active damper. Stabilizability is also important in human balancing, where the loss of balance is often associated with the loss of stabilizability of the mathematical model [27, 28]. Note that the injuries related to the death of humans are often caused by the loss of balance among the elderly. This topic has been receiving more and more attention due to the increasing average age of the world's population. Furthermore, the understanding of the human balancing process can also contribute to the field of bio-inspired robotics which has become a hot topic of research in the recent decade. This thesis presents a new delayed feedback control rule for the modeling of the human balancing process and determines the stabilizability properties of the resulting mathematical model.

1.1 Outline of the thesis

This work deals with the numerical stability analysis and the stabilizability of dynamical systems governed by delay-differential equations (DDEs). In the next subsection, the mathematical basis is briefly presented, which is necessary for the stability analysis of linear DDEs.

In Chapter 2, two novel numerical methods are described for the finite dimensional approximation of DDEs: the pseudospectral tau (PsT) and spectral element (SE) methods. The derivation of these methods are presented with computational examples and a comparison is carried out with well-known numerical methods from the literature having high convergence rates. Furthermore, the extension of the PsT and the SE methods is presented for hybrid feedback systems subjected to delay and numerical integration in the feedback loop.

In Chapter 3, the presented numerical methods are applied to different machine tool chatter models. These mathematical models take into account the feedback loop of the linear drive of the tool holder and the workpiece holder, and the feedback loop of the active damper placed close to the tool tip. Stability diagrams are determined in the plane of machining parameters and the effect of control parameters on the stability of machining is demonstrated.

In Chapter 4, the stabilizability of delayed dynamical systems is investigated. Two particular problems are studied: the optimization of control parameters for the increase of maximum admissible depth of cut in turning processes subjected to digitally controlled active damper and the loss of balance in human balancing where

the balancing process is modeled with a delayed proportional-integral-derivative-acceleration (PIDA) feedback controller.

1.2 Mathematical preliminaries

DDEs are differential equations where the argument of state terms may incorporate delays. The general form of nonlinear DDEs can be given as

$$\dot{\mathbf{x}}(t) = \mathcal{F}(t, \mathbf{x}_t), \quad (1.1)$$

where $\mathbf{x} \in \mathbb{R}^s$, $s \in \mathbb{Z}^+$ and $\mathbf{x}_t \in \mathcal{X}$ denotes a solution segment, defined by the shift

$$\mathbf{x}_t(\theta) = \mathbf{x}(t + \theta), \quad \theta \in [-\tau, 0], \quad (1.2)$$

with $\mathcal{X} = L_2([-\tau, 0]; \mathbb{R}^s)$ being the Hilbert space of square-integrable \mathbb{R}^s -valued functions on $[-\tau, 0]$. The reason for the choice of this function space is that the numerical methods, introduced in Chapter 2, define the inner product in the space of square-integrable functions. Furthermore, \mathcal{X} is required to be a Hilbert space in order to allow the tools of calculus, applied in Chapter 2. Functional $\mathcal{F}(t, \mathbf{x}_t) : \mathbb{R}^+ \times \mathcal{X} \rightarrow \mathbb{R}^s$ is a time-dependent nonlinear mapping of \mathbf{x}_t onto an s -dimensional space of real numbers. Note that the evolution of (1.1) is determined by a functional using solution segment \mathbf{x}_t which contains the history of the state \mathbf{x} dating back to time instant $t - \tau$. Consequently, for the solution of (1.1) an initial solution segment $\mathbf{x}_0 \in \mathcal{X}$ is necessary. If \mathbf{x}_0 is given, then solution segment \mathbf{x}_t can be determined at any $t > 0$ by a solution operator $\mathcal{U}(t) : \mathbb{R}^+ \times \mathcal{X} \rightarrow \mathcal{X}$ defined as

$$\mathbf{x}_t = \mathcal{U}(t)\mathbf{x}_0. \quad (1.3)$$

When $\mathcal{F}(t, \mathbf{x}_t)$ is linear then $\mathcal{U}(t)$ is also linear. If a stationary solution $\bar{\mathbf{x}}_t$ of (1.1) is known then a DDE for perturbation $\xi_t = \mathbf{x}_t - \bar{\mathbf{x}}_t$ can be derived from (1.1) in the form

$$\dot{\xi}(t) = \mathcal{L}(t)\xi_t + \mathcal{V}(t, \xi_t), \quad (1.4)$$

where $\mathcal{L}(t)$ is a time-dependent, linear functional, while $\mathcal{V}(t, \xi_t)$ is a time-dependent nonlinear functional containing all the higher-order terms of ξ_t . By the omission of higher-order terms, the linearized (or variational) system of (1.1) can be obtained in the form

$$\dot{\xi}(t) = \mathcal{L}(t)\xi_t, \quad (1.5)$$

which determines the local behavior of (1.1) close to the stationary solution $\bar{\mathbf{x}}_t$. A general form of $\mathcal{L}(t)$ which includes distributed delays and point delays can be given in the form

$$\mathcal{L}(t)\xi_t = \mathbf{A}(t)\xi_t + \sum_{p=1}^r \mathbf{B}_p(t)\xi(t - \tau_p(t)) + \int_{-\sigma}^0 \gamma(t, \theta)\xi(t + \theta) d\theta, \quad (1.6)$$

where $\mathbf{A}(t), \mathbf{B}_p(t) : \mathbb{R}^+ \rightarrow \mathbb{R}^{s \times s}$ and the kernel function $\gamma(t, \theta) : \mathbb{R}^2 \rightarrow \mathbb{R}^{s \times s}$ may contain finite number of discontinuities, thus the integral term can be rewritten as

$$\int_{-\sigma}^0 \gamma(t, \theta)\xi(t + \theta) d\theta = \sum_{b=1}^m \int_{-\sigma_{b-1}}^{-\sigma_b} \gamma_b(t, \theta)\xi(t + \theta) d\theta, \quad (1.7)$$

with continuous kernel functions $\gamma_b(t, \theta)$ and $\sigma_0 = \sigma > \sigma_1 > \dots > \sigma_m = 0$. The maximum value of the delay is denoted by $\tau = \max \left(\sup \{ \tau_p(t) \}_{p=1}^r, \sigma \right)$ and it is assumed that $\tau_p(t) > 0 \forall t$ and that τ is a finite real number.

1.2.1 Stability of linear DDEs

This thesis deals with the analysis of linear time-periodic DDEs, that is with (1.5), under the assumption that $\mathcal{L}(t) = \mathcal{L}(t + T)$ holds for all t , where $T > 0$ is the principal period of the system. In (1.6), this assumption gives $\mathbf{A}(t) = \mathbf{A}(t + T)$, $\mathbf{B}_p(t) = \mathbf{B}_p(t + T)$ and $\gamma_b(t, \theta) = \gamma_b(t + T, \theta) \forall t$. When (1.5) is time-periodic, its stability is determined by the Floquet theory. According to this theory, non-trivial solutions of (1.5) can be formulated as

$$\xi(t) = \mathbf{p}(t)e^{\lambda t}, \quad (1.8)$$

where $\lambda \in \mathbb{C}$ is called characteristic exponent and $\mathbf{p}(t) : \mathbb{R}^+ \rightarrow \mathbb{C}^s$, with $\mathbf{p}(t) = \mathbf{p}(t + T) \forall t$. This implies

$$\xi_t(\theta) = \mathbf{p}_t(\theta)e^{\lambda(t+\theta)}, \quad (1.9)$$

and

$$\xi_0(\theta) = \mathbf{p}_0(\theta)e^{\lambda\theta}, \quad (1.10)$$

$$\xi_T(\theta) = \mathbf{p}_T(\theta)e^{\lambda(T+\theta)} = \mathbf{p}_0(\theta)e^{\lambda T}e^{\lambda\theta}. \quad (1.11)$$

After substituting $\mathbf{x}(t)$ with $\xi(t)$ in (1.3) and substituting (1.10)–(1.11) to (1.3), one obtains the infinite-dimensional eigenvalue-eigenvector problem

$$(\mathcal{U}(T) - \mu^* \mathcal{I}) \xi_0 = \mathbf{0}, \quad (1.12)$$

where $\mathcal{U}(T)$ is the monodromy operator, $\mu^* = e^{\lambda T}$ is one of its eigenvalues and ξ_0 is the corresponding eigenfunction (infinite dimensional eigenvector). Since no characteristic exponent λ exists for $\mu^* = 0$, the zero eigenvalues of $\mathcal{U}(T)$ are omitted and the nonzero eigenvalues are called characteristic multipliers and are denoted by μ in the following. The necessary and sufficient condition for the stability of (1.5) with time-periodic $\mathcal{L}(t)$ is that all characteristic multipliers have modulus less than one. Note that for any characteristic exponent the corresponding characteristic multiplier can be uniquely determined. However, this is not true in the opposite direction, multipliers do not uniquely determine exponents.

A special case of time-periodic DDEs are the autonomous DDEs, where the functional \mathcal{L} has no time-dependency, that is \mathbf{A} , \mathbf{B}_p and γ matrices and τ_p delays are time-independent in (1.6). Therefore, in case of autonomous systems, the principal period is not defined and it can be chosen arbitrarily as any finite $T \in \mathbb{R}^+$ number, hence the stability can be still analyzed using the monodromy operator $\mathcal{U}(T)$. In the autonomous case \mathbf{p} is time-independent in the non-trivial solutions (1.8) of (1.5). After the substitution of (1.8) to (1.5)–(1.6), one arrives at the characteristic equation

$$\det \left(\lambda \mathbf{I} - \mathbf{A} - \sum_{p=1}^v \mathbf{B}_p e^{-\lambda \tau_p} - \int_{-\sigma}^0 \gamma(\theta) e^{-\lambda \theta} d\theta \right) = 0. \quad (1.13)$$

The roots of this characteristic equation are the characteristic exponents. In case of autonomous DDEs, the condition of exponential stability is that all the infinitely many characteristic exponents have negative real parts. Consequently, the stability of autonomous DDEs can be determined either by calculating the critical eigenvalues of $\mathcal{U}(T)$ (i.e, the characteristic multipliers) or by calculating the critical roots of the characteristic equation (1.13) (i.e, the critical characteristic exponents).

1.2.2 Equivalent forms of delayed systems

Time-delay problems can be formulated not only in the form of DDEs. In fact, other mathematical problems can be constructed which are equivalent to (1.5) in their primary characteristics.

Let operator $\mathcal{V} : \mathcal{D}(\mathcal{V}) \rightarrow \mathcal{X}^\pm$ be defined as

$$\mathcal{V}(\xi_0, \xi^+)(\theta) = \begin{cases} \xi_0(\theta) & \text{if } \theta \in [-\tau, 0], \\ \xi^+(\theta) & \text{if } \theta \in (0, h], \end{cases} \quad (1.14)$$

with domain

$$\mathcal{D}(\mathcal{V}) = \{\xi_0 \in \mathcal{X}, \xi^+ \in \mathcal{X}^+ : \xi_0(0) = \xi^+(0)\}. \quad (1.15)$$

Here $h > 0$ and $\mathcal{X}^\pm \subset L_2([-\tau, h], \mathbb{R}^s)$, while $\mathcal{X}^+ = L_2^1([0, h], \mathbb{R}^s)$ is the Hilbert space of \mathbb{R}^s -valued functions which are square-integrable and whose first derivative is also square-integrable on interval $[0, h]$. Note that operator $\mathcal{V}(\xi_0, \xi^+)$ simply connects the initial function segment ξ_0 and function segment ξ^+ at $\theta = 0$. Now considering the residual of (1.5) on $t \in [0, h]$, one can construct the operator equation (OpE)

$$\mathcal{A}\mathbf{z} = \mathbf{0}, \quad (1.16)$$

where operator $\mathcal{A} : \mathcal{X}^\pm \rightarrow L_2([0, h], \mathbb{R}^s)$ is defined by

$$\mathcal{A}\mathbf{z} = \{\dot{\mathbf{z}}(t) - \mathcal{L}(t)\mathbf{z}_t : t \in [0, h]\}. \quad (1.17)$$

Since $\mathcal{L}(t)$ is linear, one can decompose (1.16) by plugging (1.14) into (1.16) as

$$\mathcal{A}^-\xi_0 + \mathcal{A}^+\xi^+ = \mathbf{0}, \quad (1.18)$$

where

$$\mathcal{A}^-\xi_0 = \mathcal{A}\mathcal{V}(\xi_0, \mathbf{0}), \quad \mathcal{A}^+\xi^+ = \mathcal{A}\mathcal{V}(\mathbf{0}, \xi^+). \quad (1.19)$$

For any $\xi_0 \in \mathcal{X}$ initial function segment, the solution ξ^+ of (1.18) is precisely defined by $\xi^+ = \{\xi(t) : t \in [0, h]\}$. When $\mathcal{L}(t)$ is time-periodic and $h = T$, the monodromy operator $\mathcal{U}(T)$ can be expressed using (1.18) (see Section 2.2). Therefore, (1.18) can be used for the stability analysis of (1.5).

Stability properties of (1.5) can also be described by the operator differential equation (OpDE)

$$\dot{\xi}_t = \mathcal{G}(t)\xi_t, \quad (1.20)$$

where operator $\mathcal{G} : \mathcal{D}(\mathcal{G}) \rightarrow \mathcal{X}$ is given by

$$\mathcal{G}(t)\xi_t = \xi'_t, \quad (1.21)$$

with domain $\mathcal{D}(\mathcal{G}) = \mathbb{R}^+ \times \mathcal{Y}$, where

$$\mathcal{Y} = \{\xi_t \in \mathcal{X} : \xi'_t \in \mathcal{X}, \xi'_t(0) = \mathcal{L}(t)\xi_t\} \subseteq \mathcal{X} \quad (1.22)$$

and $\xi'_t(\theta)$ is the derivative of $\xi_t(\theta)$ with respect to θ . Note that \mathcal{G} cannot be correctly defined when \mathcal{Y} is time-dependent, hence varying (see page 341 of [16]). Therefore, we assume that \mathcal{Y} can be selected as a time-independent domain (the maximum delay τ is finite). When \mathcal{G} is time-invariant then (1.20) gives an abstract Cauchy problem and operator \mathcal{G} is called its infinitesimal generator. If and only if all the elements of the spectrum of the infinitesimal generator lie on the left half of the complex plane, then the abstract Cauchy problem is asymptotically stable.

In Chapter 7.1 of [25], Lemma 1.2. shows that the solution of the abstract Cauchy problem is defined by the solution operator $\mathcal{U}(t)$. Furthermore, in Chapter 7.2 of [25], it is also shown that the spectrum of \mathcal{G} is precisely given by the roots of the characteristic equation of (1.5). This latter implies that, for the autonomous case, the stability analysis of (1.5) and (1.20) give the same results. Numerical results presented in [13] show that (1.5) and (1.20) are equivalent regarding stability for time-periodic systems as well. Note, that the conversion of (1.5) to (1.20) is similar to the conversion of a high-order ordinary differential equation (ODE) to first-order ODEs (Cauchy normal form).

By the introduction of function $\mathbf{y}(t, \theta) = \xi_t(\theta)$ with two independent variables t and θ , (1.20) gives the hyperbolic partial differential equation (PDE)

$$\frac{\partial \mathbf{y}(t, \theta)}{\partial t} = \frac{\partial \mathbf{y}(t, \theta)}{\partial \theta}, \quad \theta \in [-\tau, 0], \quad (1.23)$$

with the linear, time-dependent, non-local boundary condition

$$\left. \frac{\partial \mathbf{y}(t, \theta)}{\partial t} \right|_{\theta=0} = \mathcal{L}(t)\mathbf{y}(t, \theta). \quad (1.24)$$

This PDE representation is equivalent to (1.20) and it is often used in the literature to describe delayed systems [86, 87]. Note, that in some engineering applications the governing equation which models the physical phenomenon can be derived directly in the form (1.23)–(1.24), instead of (1.5). For example, the widely used DDE model of turning with constant delay (see Chapter 5.1.2 in [29]) is a special case of the PDE model, introduced in [88]. In particular, the DDE model describes the cutting tool's motion in the PDE model under the condition that the tool never loses contact with the workpiece (for details see [50]). In general, equations (1.23)–(1.24) are capable of giving a more detailed description of the delayed system than equation (1.5). This is due to the fact that in the PDE model, (1.23) describes the propagation of information while (1.24) introduces delays in the system. In contrast, DDE (1.5) embeds the propagation in the time domain using time lags. During mathematical modeling, a decision between the use of (1.23)–(1.24) or (1.5) always involves a trade-off since, in general, the analysis of PDEs is more difficult than the analysis of DDEs.

Chapter 2

Numerical methods for stability analysis

This chapter presents two numerical methods with high convergence rates for the numerical stability analysis of delay-differential equations (DDEs). In addition to the derivation of these numerical methods they are also compared to recently developed methods from the engineering literature. Furthermore, the extension of these two numerical methods is presented for hybrid systems which include additional terms with piecewise continuous arguments. The two numerical methods both employ the method of weighted residuals in order to obtain a finite dimensional approximation of the infinite dimensional problem of DDEs.

In the following, the method of weighted residuals is briefly demonstrated on (1.20). At time instant t , one can approximate the solution segment $\xi_t(\theta)$ of (1.20) on the domain $\theta \in [-\tau, 0]$, using finite number of unknown variables $\mathbf{a}_j(t)$ in a finite dimensional function space spanned by the basis $\{\phi_j\}_{j=1}^{n+1}$. The approximate solution of (1.20) therefore has the form

$$\tilde{\xi}_t(\theta) = \sum_{j=1}^{n+1} \mathbf{a}_j(t) \phi_j(\theta). \quad (2.1)$$

After the substitution of $\tilde{\xi}_t$ into (1.20), one obtains the residual function

$$\mathbf{r}_t(\theta) = \sum_{j=1}^{n+1} \dot{\mathbf{a}}_j(t) \phi_j(\theta) - \sum_{j=1}^{n+1} \mathbf{a}_j(t) \phi'_j(\theta) \neq \mathbf{0}, \quad \theta \in [-\tau, 0]. \quad (2.2)$$

Note that in general the residual function is not zero since $\tilde{\xi}_t$ is only an approximation of ξ_t . Approximation schemes aim to determine coefficients $\mathbf{a}_j(t)$ in a way that the approximate solution segment $\tilde{\xi}_t$ would be closest to the exact solution of (1.20). The method of weighted residuals weights the residual function $\mathbf{r}_t(\theta)$ by test functions $\psi_i(\theta)$ over the domain $\theta \in [-\tau, 0]$ in order to obtain a set of linearly independent equations, from which coefficients $\mathbf{a}_j(t)$ can be determined. The application of the method of weighted residuals to (2.2), gives

$$\langle \mathbf{r}_t, \psi_i \rangle = 0, \quad i = 1, 2, \dots, n+1; \quad (2.3)$$

where the inner product of functions $\mathbf{r}_t(\theta)$ and $\psi_i(\theta)$ is defined according to

$$\langle \mathbf{r}_t, \psi_i \rangle = \int_a^b \mathbf{r}_t(\theta) \psi_i(\theta) d\theta, \quad (2.4)$$

with $\theta \in [a, b] = [-\tau, 0]$ being the domain of functions $\mathbf{r}_t(\theta)$ and $\psi_i(\theta)$. Equations (2.3) can be represented in a matrix form as

$$\mathbf{N}\dot{\mathbf{a}}(t) = \mathbf{M}\mathbf{a}(t), \quad (2.5)$$

where matrices $\mathbf{N} \in \mathbb{R}^{s(n+1) \times s(n+1)}$ and $\mathbf{M} \in \mathbb{R}^{s(n+1) \times s(n+1)}$ are composed from sub-matrices according to

$$\mathbf{N} = [\langle \phi_j, \psi_i \rangle \mathbf{I}]_{i,j=1}^{n+1,n+1}, \quad \mathbf{M} = [\langle \phi'_j, \psi_i \rangle \mathbf{I}]_{i,j=1}^{n+1,n+1}, \quad (2.6)$$

while $\mathbf{I} \in \mathbb{R}^{s \times s}$ is an identity matrix and $\mathbf{a}(t) = [\mathbf{a}_j(t)]_{j=1}^{n+1}$. Note that due to the application of the method of weighted residuals the state space \mathcal{X} has to be a Hilbert space, however the proper selection of \mathcal{X} falls out of the scope of this thesis. Note also that the solution of (2.3) is not in the domain $\mathcal{D}(\mathcal{G})$ since boundary condition

$$\tilde{\xi}'_t(0) = \mathcal{L}(t)\tilde{\xi}_t \quad (2.7)$$

is not satisfied. For the partial differential equation (PDE) representation this means that (1.23) is employed but (1.24) is not satisfied. Consequently, in order to approximate (1.20), boundary condition (2.7) has to be enforced.

Based on the above description, weighted residual type methods can differ in the way they select the set of base functions and the set of test functions and also in the way they enforce the boundary conditions. Methods thus can be categorized based on how they enforce the boundary conditions. There are two main categories which are briefly discussed below.

The Galerkin approximation considers the boundary conditions as constraints on the approximate solution $\tilde{\xi}_t$, that is, base functions ϕ_j are constructed in a way that $\tilde{\xi}_t$ satisfies the boundary condition. Note, however that boundary condition (2.7) is non-local, that is, it requires the exact solution of (1.5) to be known. Thus, the non-locality of boundary condition (2.7) implies that Galerkin methods cannot be used for the approximation of (1.20).

The tau approximation can be used to solve the problem with non-local boundary conditions. Here, in contrast with the Galerkin method, base functions ϕ_j do not need to satisfy the boundary constraints. The tau approximation technique simply replaces an equation from (2.3) by the discretized boundary condition (2.7). This replacement relies on the tau method which was proposed by Lanczos (see [41]). The tau method claims that if, after this replacement, (2.3) still defines a proper projection then the approximate solution is an element of a complete finite dimensional subspace of the solutions of the original problem (1.20). Note again, that in this case \mathcal{X} has to be a proper Hilbert space. More precise details on Galerkin and tau approximations can be found in Chapter 2 of [23].

2.1 Pseudospectral tau method

For the pseudospectral tau (PsT) method, the term "pseudospectral" indicates that the solution is approximated in a finite dimensional subspace, where the set of basis functions $\{\phi_j\}_{j=1}^{n+1}$ are chosen in a way that the coordinates $\mathbf{a}_j(t)$ of the subspace spanned by $\{\phi_j\}_{j=1}^{n+1}$ represent the approximate solution $\tilde{\xi}_t(\theta)$ at specific points of

$\theta \in [-\tau, 0]$. In the following, the approximation concept of the PsT method is given in details.

2.1.1 Lagrange interpolation

The PsT method approximates the solution segment by its Lagrange interpolant. The approximate solution segment is given in the form

$$\tilde{\xi}_t(\theta) = \sum_{j=1}^{n+1} \phi_j(\theta) \xi_t(\theta_j), \quad \theta \in [-\tau, 0], \quad (2.8)$$

where $\theta_j \in [-\tau, 0]$ are the nodes of interpolation and $\phi_j \in \mathcal{P}_n$ are the Lagrange base polynomials, with \mathcal{P}_n denoting the space of polynomials of order n . Note that by using interpolant (2.8), the unknown coefficients of (2.1) become particular values of the approximate solution segment at some points θ_j , that is $\mathbf{a}_j(t) = \xi_t(\theta_j)$. Lagrange base polynomials have the property

$$\phi_j(\theta_k) = \delta_{j,k}, \quad (2.9)$$

where $\delta_{j,k}$ denotes the Kronecker-delta function. The classical form of Lagrange base polynomials is

$$\phi_j(\theta) = \prod_{\substack{k=1 \\ k \neq j}}^{n+1} \frac{\theta - \theta_k}{\theta_j - \theta_k}. \quad (2.10)$$

However, the above formula has some disadvantages: it needs high number of floating point operations to calculate ϕ_j at any given point other than the nodes of interpolation, while the rounding errors can lead to numerical instability, furthermore the formula for the derivative of ϕ_j is very complicated. The so-called barycentric representation of Lagrange interpolants helps in the above problems. The barycentric formula of Lagrange base polynomials is defined by

$$\phi_j(\theta) = \frac{\varpi_j}{\sum_{k=1}^{n+1} \frac{\varpi_k}{\theta - \theta_k}}, \quad (2.11)$$

where the barycentric weights are given as

$$\varpi_j = \frac{1}{\omega'(\theta_j)}, \quad \omega(\theta) = \prod_{j=1}^{n+1} (\theta - \theta_j). \quad (2.12)$$

At the nodes of interpolation the derivatives of the Lagrange base polynomials can be calculated as

$$\phi'_j(\theta_k) = \begin{cases} \frac{\varpi_j / \varpi_k}{\theta_k - \theta_j} & j \neq k, \\ -\sum_{\substack{q=1 \\ q \neq j}}^{n+1} \frac{\varpi_q / \varpi_j}{\theta_j - \theta_q} & j = k. \end{cases} \quad (2.13)$$

Details and derivation of the above formulae can be found in [7]. Note, that using (2.13), the derivative of the Lagrange interpolant at the nodes of interpolation can

be calculated simply by a matrix-vector multiplication, since

$$\tilde{\xi}'_t(\theta_j) = \sum_{j=1}^{n+1} D_{k,j} \tilde{\xi}_t(\theta_j), \quad k = 1, \dots, n+1; \quad (2.14)$$

where $D_{k,j} = \phi'_j(\theta_k)$. Note that the value of the Lagrange interpolant at $n+1$ number of arbitrary distinct points $\{\theta_l^*\}_{l=1}^{n+1}$ is given by a matrix-vector multiplication:

$$\tilde{\xi}_t(\theta_l^*) = \sum_{j=1}^{n+1} L_{l,j} \tilde{\xi}_t(\theta_j), \quad l = 1, \dots, n+1; \quad (2.15)$$

where $L_{l,j} = \phi_j(\theta_l^*)$ defines a linear transformation between the two point sets given by the interpolant evaluated at node sets $\{\theta_l^*\}_{l=1}^{n+1}$ and $\{\theta_j\}_{j=1}^{n+1}$. Due to the uniqueness of the Lagrange interpolant, its derivative can be calculated at any given point set $\{\theta_l^*\}_{l=1}^{n+1}$ by the matrix-matrix-vector multiplication

$$\tilde{\xi}'_t(\theta_l^*) = \sum_{k=1}^{n+1} \sum_{j=1}^{n+1} L_{l,k} D_{k,j} \tilde{\xi}_t(\theta_j), \quad l = 1, \dots, n+1. \quad (2.16)$$

2.1.2 Tau approximation

Using (2.8), after the replacement of the equation corresponding to $i = n+1$ in (2.3) with the discretized boundary condition (2.7) and coordinate transformation $\zeta = 2\theta/\tau + 1$, equation (2.5) obtains the form

$$\mathbf{N} \dot{\mathbf{X}}(t) = \mathbf{M}(t) \mathbf{X}(t), \quad (2.17)$$

where $\mathbf{X}(t) = [\tilde{\xi}_t(\zeta_j)]_{j=1}^{n+1}$ while the elements of matrices \mathbf{N} and $\mathbf{M}(t)$ are now given according to

$$\mathbf{N}_{i,j} = \begin{cases} \langle \phi_j, \psi_i \rangle \mathbf{I} & i = 1, \dots, n; \\ \phi_j(1) \mathbf{I} & i = n+1; \end{cases} \quad (2.18)$$

$$\mathbf{M}_{i,j}(t) = \begin{cases} \frac{2}{\tau} \langle \phi'_j, \psi_i \rangle \mathbf{I} & i = 1, \dots, n; \\ \mathcal{L}(t) \phi_j & i = n+1. \end{cases} \quad (2.19)$$

If \mathbf{N} is invertible then (2.17) can be written as

$$\dot{\mathbf{X}}(t) = \mathbf{G}(t) \mathbf{X}(t), \quad (2.20)$$

where $\mathbf{G}(t) = \mathbf{N}^{-1} \mathbf{M}(t)$ is a finite dimensional approximation of operator $\mathcal{G}(t)$. Note that in (2.3) the multiplication of residual function (2.2) with test functions ψ_i , $i = 1, \dots, n$ gives the projection of the residual function onto a space spanned by the test function set $\{\psi_i\}_{i=1}^n$. Since the approximate solution (2.8) is an element of the subspace of polynomials of order n , by choosing the test function set as a base in the subspace of polynomials of order n , the residual (2.2) will be zero in a co-dimension one subspace of the space \mathcal{P}_n of (2.8). The boundary condition is enforced by (2.7), which gives additional relationship between coefficients $\tilde{\xi}_t(\theta_j)$. If a proper set of test functions is chosen, the solution of the system subject to tau approximation can be an element of a complete finite dimensional subspace of the

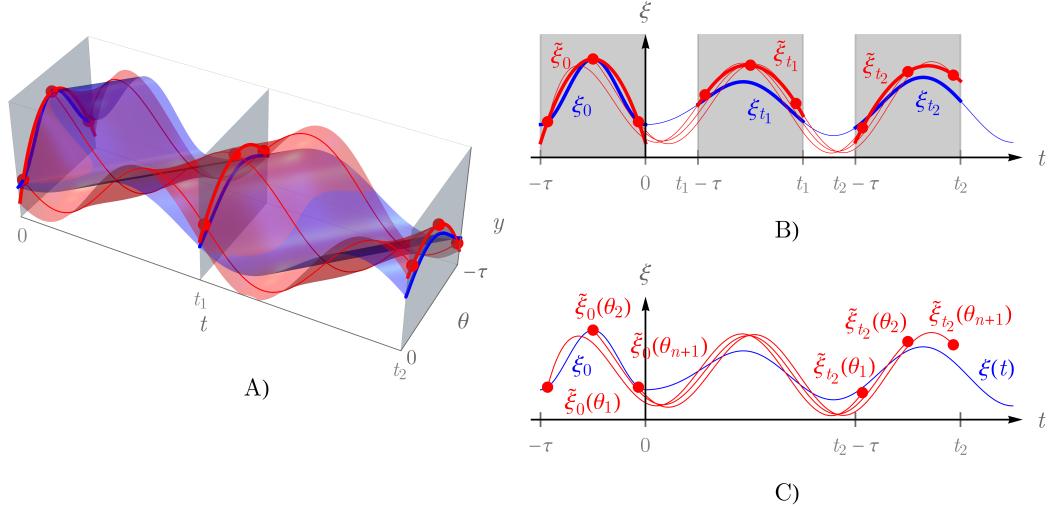


FIGURE 2.1: Representation of the solution A) of PDE (1.23)–(1.24) (blue surface), B) of OpDE (1.20) (thick blue lines) and C) of DDE (1.5) (blue line), and their ODE approximations (red surface and lines).

solution of the original problem (1.20). Consequently, by increasing n the approximation can converge to the original problem. In this thesis, the suitability of the test function set is not investigated analytically, but they have been tested numerically. The numerical experiments showed that for the autonomous case the eigenvalues of \mathbf{G} in (2.20) give a finite dimensional approximation for the point spectrum of \mathcal{G} .

Although this method is capable of the approximation of the point spectrum of \mathcal{G} , the main focus of this thesis is only on the numerical stability analysis of DDEs. Since the ordinary differential equation (ODE) approximation (2.20) is stable if and only if all eigenvalues of \mathbf{G} are located in the left half of the complex plane, only the rightmost eigenvalue of \mathbf{G} is of interest. Note that if DDE (1.5) is modified that affects only the last rows of $\mathbf{M}(t)$ corresponding to $i = n + 1$ and the multiplier of the remaining rows. Therefore, when s and n are fixed, the inverse of \mathbf{N} and the integral terms in (2.18)–(2.19) have to be calculated only once, which is beneficial during the construction of stability charts (as discussed in Section 2.3).

In Figure 2.1, an illustration of the ODE approximation (2.20) is shown for $s = 1$ and $n = 2$. The solution of the ODE approximation is presented in Figures 2.1/A and 2.1/B together with the solution for the PDE (1.23)–(1.24) and operator differential equation (OpDE) (1.20) representations of DDE (1.5), respectively. In Figure 2.1, the exact solution is shown by blue color while the solution of the ODE approximation is depicted by red color. The thin red lines are given by the points of the interpolant at the nodes of interpolation, that is they are given by the elements of $\mathbf{X}(t)$. In Figure 2.1/A, these lines define the red surface which approximates the exact solution of (1.23)–(1.24). Note that, due to (1.23), the isocurves of the blue surface are lines with slope 1. In Figure 2.1/B the thin red lines define an approximation for the solution segment ξ_t of (1.20) at each time instant t . These solution segment approximations are shown by thick red lines for time instants $0, t_1$ and t_2 . At each t time instant, the solution segment ξ_t coincides with the corresponding segment of $y(t, \theta)$. This is highlighted by gray windows in Figures 2.1/A and 2.1/B for time instants $0, t_1$ and t_2 . Note that each element of $\mathbf{X}(t)$ gives an approximation for the solution $\xi(t)$ of DDE (1.5) due to the definition (1.2) of solution segment ξ_t . This is illustrated in Figure 2.1/C.

2.1.3 Numerical integration

Equations (2.18)–(2.19) contain integral terms whose analytical evaluation would be time-consuming or even impossible. Note however, that since $\{\psi_i\}_{i=1}^n \subset \mathcal{P}_{n-1}$, the terms in the integrals are polynomials with maximum order $2n - 1$. As a result, the integral terms in (2.18)–(2.19) can be evaluated accurately by the Lobatto-type Legendre-Gauss quadrature (see Appendix A.2 for details) using $n + 1$ number of points. With the application of this quadrature, the integral terms in (2.18)–(2.19) can be calculated as

$$\langle \phi_j, \psi_i \rangle = \sum_{q=1}^{n+1} F_{i,q} L_{q,j}, \quad i = 1, \dots, n+1; \quad (2.21)$$

$$\langle \phi'_j, \psi_i \rangle = \sum_{q=1}^{n+1} \sum_{k=1}^{n+1} F_{i,q} L_{q,k} D_{k,j}, \quad i = 1, \dots, n+1; \quad (2.22)$$

where $F_{i,q} = \psi_i(\zeta_q^*) w_q$ with $\{\zeta_q^*\}_{q=1}^{n+1} \subset [-1, 1]$ and $\{w_q\}_{q=1}^{n+1}$ being the set of nodes and the corresponding set of weights of the quadrature. Note that if the interpolation and quadrature node sets are the same, then $L_{q,j} = \delta_{q,j}$, that is formulas (2.21)–(2.22) are simplified by one matrix multiplication. Also note that with the application of the standard (non-Lobatto-type) Legendre-Gauss quadrature for integration, the number of multiplications could be further decreased. This is owing to the fact that this quadrature gives accurate results for the integral of polynomials of order $2n - 1$ by using only n number of nodes.

2.1.4 Example

Consider the DDE given by (1.5)–(1.7). The substitution of (2.21)–(2.22) into (2.18)–(2.19) gives

$$\mathbf{N}_{i,j} = \begin{cases} \sum_{q=1}^{n+1} F_{i,q} L_{q,j} \mathbf{I} & i = 1, \dots, n; \\ \phi_j(1) \mathbf{I} & i = n+1; \end{cases} \quad (2.23)$$

$$\mathbf{M}_{i,j}(t) = \begin{cases} \frac{2}{\tau} \sum_{q=1}^{n+1} \sum_{k=1}^{n+1} F_{i,q} L_{q,k} D_{k,j} \mathbf{I} & i = 1, \dots, n; \\ \mathcal{L}(t)\phi_j & i = n+1. \end{cases} \quad (2.24)$$

Using Lobatto-type Legendre-Gauss quadrature for integration the last row of $\mathbf{M}_{i,j}(t)$ can be expanded as

$$\begin{aligned} \mathcal{L}(t)\phi_j &\approx \mathbf{A}(t)\phi_j(1) + \sum_{p=1}^r \mathbf{B}_p(t)\phi_j(-\hat{\tau}_p(t)) \\ &\quad + \sum_{b=1}^m \frac{\tau(\hat{\sigma}_b - \hat{\sigma}_{b-1})}{4} \sum_{q=1}^{n+1} \gamma\left(t, \frac{(\hat{\zeta}_q - 1)\tau}{2}\right) \phi_j(\hat{\zeta}_q) w_q, \end{aligned} \quad (2.25)$$

where $\hat{\sigma}_{b-1} = 1 - 2\sigma_{b-1}/\tau$, $\hat{\sigma}_b = 1 - 2\sigma_b/\tau$, $\hat{\tau}_p = -1 + 2\tau_p/\tau$ and $\hat{\zeta}_q = \frac{\hat{\sigma}_b - \hat{\sigma}_{b-1}}{2}(\zeta_q^* + 1) + \hat{\sigma}_{b-1}$, while Lagrange base polynomials $\phi_j(\zeta)$ are defined by the node set $\{\zeta_j\}_{j=1}^{n+1}$ on the rescaled domain $\zeta \in [-1, 1]$. Finally, the finite dimensional approximation of $\mathcal{G}(t)$ is given by (2.17)–(2.20).

2.1.5 Selection of node and test function sets

Note that due to the Lagrange interpolation, the base function set $\{\phi_j\}_{j=1}^{n+1}$ is precisely defined by the node set $\{\zeta_j\}_{j=1}^{n+1}$ of interpolation. Consequently, the error between the interpolated function segment ξ_t and its interpolant $\tilde{\xi}_t$ can be minimized by the proper selection of node set $\{\zeta_j\}_{j=1}^{n+1}$. The error $|\xi_t(\zeta) - \tilde{\xi}_t(\zeta)|_{C[-1,1]}$ between ξ_t and $\tilde{\xi}_t$ can be minimized by the selection of Chebyshev nodes for the nodes of interpolation, where the distance between ξ_t and $\tilde{\xi}_t$ is measured by the norm

$$|\xi_t(\theta)|_{C[a,b]} = \max \{ |\xi_t(\theta)| : \theta \in [a, b] \}. \quad (2.26)$$

Details on Chebyshev nodes and polynomials can be found in Appendix A.3. Due to its minimum error property, the Chebyshev node set is used for the PsT method as the node set of interpolation.

For the test function set $\{\psi_i\}_{i=1}^n$ two choices were investigated: the set $\{\zeta^{i-1}\}_{i=1}^n$ (which is a base in \mathcal{P}_{n-1}), and the set defined by Legendre polynomials of order up to $n-1$ (which is an orthogonal base in \mathcal{P}_{n-1}). For details on Legendre polynomials see Appendix A.1. In case of low n , these sets give precisely the same results for the eigenvalues of \mathbf{G} . However, above a certain order, the set $\{\zeta^{i-1}\}_{i=1}^n$ results in badly conditioned matrix \mathbf{N} , which destroys the convergence of the method. This problem is avoided, by using an orthogonal base in \mathcal{P}_{n-1} (e.g. the Legendre polynomials). Consequently, for the PsT method, the test functions are defined as $\psi_i = P_{i-1}$, $i = 1, \dots, n$; where P_i is a Legendre polynomial of order i .

2.1.6 Stability of time-periodic systems

For the autonomous case, the stability of ODE approximation (2.20) is determined by its rightmost eigenvalue on the complex plane of the matrix approximation \mathbf{G} of the infinitesimal generator \mathcal{G} . In contrast, for time-periodic systems a matrix approximation \mathbf{U} for the monodromy operator $\mathcal{U}(T)$ has to be determined. Since the elements of $\mathbf{X}(t)$ precisely determine the approximate solution segment $\tilde{\mathbf{x}}_t$ at $t \geq 0$ time instant, a matrix approximation of the monodromy operator can be defined by the mapping

$$\mathbf{X}(T) = \mathbf{U}\mathbf{X}(0). \quad (2.27)$$

In general, \mathbf{U} cannot be calculated from (2.20) in closed form, however a simple approximation for \mathbf{U} can be determined via the approximation of the time-periodic ODE (2.20) by a series of autonomous ODEs in the form

$$\dot{\tilde{\mathbf{X}}}^k(t) = \mathbf{G}_k \tilde{\mathbf{X}}^k(t) \quad t \in [(k-1)\Delta\tilde{t}, k\Delta\tilde{t}], \quad k = 1, 2, \dots, \tilde{m}; \quad (2.28)$$

where $\tilde{\mathbf{X}}^k = \left\{ \tilde{\mathbf{X}}^k(t) : t \in [(k-1)\Delta\tilde{t}, k\Delta\tilde{t}] \right\}$ solution segments are connected by boundary conditions

$$\tilde{\mathbf{X}}^k(k\Delta\tilde{t}) = \tilde{\mathbf{X}}^{k+1}(k\Delta\tilde{t}), \quad k = 1, 2, \dots, \tilde{m}-1. \quad (2.29)$$

In (2.28), the time step is $\Delta\tilde{t} = T/\tilde{m}$ with $\tilde{m} \in \mathbb{N}$ being the period resolution, while

$$\mathbf{G}_k = \frac{1}{\Delta\tilde{t}} \int_{(k-1)\Delta\tilde{t}}^{k\Delta\tilde{t}} \mathbf{G}(\theta) d\theta \quad (2.30)$$

is a piecewise constant approximation of $\mathbf{G}(t)$. The approximation of \mathbf{U} is then given by

$$\mathbf{U} \approx e^{\hat{\mathbf{G}}_{\tilde{m}}} e^{\hat{\mathbf{G}}_{\tilde{m}-1}} \cdots e^{\hat{\mathbf{G}}_1}, \quad (2.31)$$

where

$$\hat{\mathbf{G}}_k = \mathbf{G}_k \Delta \tilde{t} = \mathbf{N}^{-1} \int_0^{\Delta \tilde{t}} \mathbf{M}(\theta + (k-1)\Delta \tilde{t}) d\theta, \quad k = 1, 2, \dots, \tilde{m}. \quad (2.32)$$

Note that the monodromy operator is approximated in two steps. First, (1.20) is discretized by the PsT method, then (2.20) is approximated by (2.28). In contrast, the direct approximation of the solution operator is more sound, which can be carried out e.g. by the spectral element (SE) method or the collocation technique in [10].

In the following, two examples are shown for the calculation of stability charts of time-periodic DDEs. Namely, stability charts are determined for the delayed Mathieu equation and for an oscillator with time-periodic delay. Examples for the calculation of stability charts of autonomous DDEs are presented in Section 2.3.

Delayed Mathieu equation. The first-order form of the delayed Mathieu equation is

$$\dot{\xi}(t) = (\mathbf{A}_0 + \mathbf{A}_T(t)) \xi(t) + \mathbf{B} \xi(t - \tau), \quad (2.33)$$

where

$$\mathbf{A}_0 = \begin{bmatrix} 0 & 1 \\ -a & 0 \end{bmatrix}, \quad \mathbf{A}_T(t) = \varepsilon \begin{bmatrix} 0 & 0 \\ -\cos(\frac{2\pi t}{T}) & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ b & 0 \end{bmatrix}. \quad (2.34)$$

The derivation of the exact stability boundaries for this equation can be found in [31]. Using the PsT method, the discretization of (2.33) is carried out according to (2.23)–(2.24), where now

$$\mathcal{L}(t)\phi_j = (\mathbf{A}_0 + \mathbf{A}_T(t))\phi_j(1) + \mathbf{B}\phi_j(-1). \quad (2.35)$$

Note, that for the computation of \mathbf{U} , integral terms in (2.32) have to be calculated only once, even for varying a, b, ε, τ system parameters. By increasing both n and \tilde{m} , one can observe the convergence of the approximate stability chart to the analytical one. In Figure 2.2, the converged stability boundaries are shown for fixed τ and T parameters.

Oscillator with time-periodic delay. The first-order form of this oscillator is

$$\dot{\xi}(t) = \mathbf{A}_0 \xi(t) + \mathbf{B} \xi(t - \tau_1(t)), \quad (2.36)$$

where matrices \mathbf{A}_0 and \mathbf{B} are defined in (2.34), and $\tau_1(t+T) = \tau_1(t)$ with $\tau_1(t) \geq 0 \ \forall t$. Again, the discretization of (2.36) is performed according to (2.23)–(2.24), where now

$$\mathcal{L}(t)\phi_j = \mathbf{A}_0 \phi_j(1) + \mathbf{B} \phi_j\left(1 - \frac{2\tau_1(t)}{\tau}\right), \quad (2.37)$$

with τ being the maximum value of the delay. In Figure 2.3 approximate stability charts are shown for the time-periodic delay

$$\tau_1(t) = 2\pi \left(1 + \varepsilon \cos\left(\frac{2\pi t}{T}\right)\right). \quad (2.38)$$

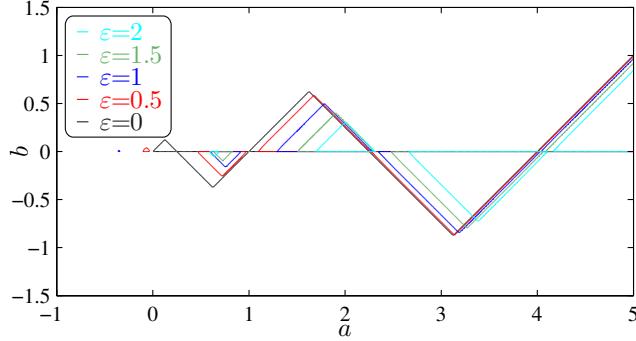


FIGURE 2.2: Stability chart of the delayed Mathieu equation for $\tau = T = 2\pi$, $n = 12$ and $\tilde{m} = 15$

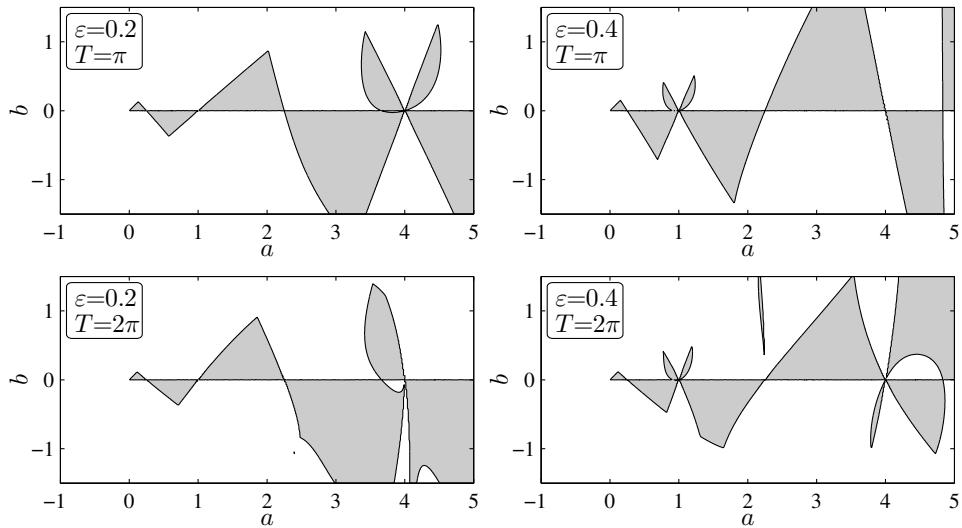


FIGURE 2.3: Stability charts of the oscillator with time-periodic delay for $n = 15$ and $\tilde{m} = 20$.

After the comparison of the results presented in Figure 2.3 and the results shown in Fig. 4.11 in Chapter 4 of [29], one can observe that the boundaries of stability, determined by the PsT method, converge to those determined by the semi-discretization method.

2.2 Spectral element method

The SE method has been recently introduced in the literature for the stability analysis of DDEs [36–38]. This numerical method was first published in [37] for autonomous DDEs with single delay, thereafter it was extended in [38] to autonomous DDEs with distributed delay and generalized to time-periodic DDEs with multiple point delays in [36]. Although the spectral element method has been a handy tool for the stability analysis of time-periodic DDEs, its application was not straightforward due to the lack of explicit formulas for the computation of the matrix approximation \mathbf{U} of monodromy operator $\mathcal{U}(T)$. While explicit formulas were presented in [37] for the construction of \mathbf{U} for autonomous DDEs with a single point delay, [38] and [36] do not provide any explicit formula for the general case with multiple

point and distributed delays and with time-periodic coefficients. In the following, the derivation of the SE method is given for (1.5)–(1.7) based on [48]. Explicit formulas are derived for all components of the matrices, which are necessary for the computation of \mathbf{U} . Furthermore, in contrast with [36], here the formula of \mathbf{U} is derived using operator equations.

By assuming constant delays and with the application of numerical integration, (1.5)–(1.7) can be approximated by a DDE with point delays in the form

$$\dot{\xi}(t) = \mathbf{A}(t)\xi(t) + \sum_{p=1}^v \mathbf{B}_p(t)\xi(t - \tau_p), \quad (2.39)$$

where $v > r$ and for $p > r$

$$\mathbf{B}_p(t) = \frac{\sigma_{b(p)-1} - \sigma_{b(p)}}{2} \gamma_{b(p)} \left(t, \theta_{q(p)}^* \right) w_{q(p)}, \quad (2.40)$$

with delays

$$\tau_p = -\theta_{q(p)}^* = \frac{\sigma_{b(p)} - \sigma_{b(p)-1}}{2} \left(\zeta_{q(p)}^* + 1 \right) + \sigma_{b(p)-1} \quad (2.41)$$

and indices

$$q(p) = \text{mod}(p - r - 1, n + 1) + 1 \quad (2.42)$$

$$b(p) = \text{floor} \left(\frac{p - r - 1}{n + 1} \right) + 1. \quad (2.43)$$

In the above formulas $\text{mod}(a, b)$ is the modulo function of a with respect to b , while $\text{floor}(\cdot)$ denotes the floor function. Similarly to the PsT method, $\zeta_q^* \in [-1, 1]$ and w_q denote the quadrature nodes and the corresponding weights. By using $n + 1$ number of quadrature nodes for each integral term $v = r + m(n + 1)$. The SE method applies the Lobatto-type Legendre-Gauss quadrature because it keeps a good accuracy and the method reuses the quadrature during the calculation. Note however, that quadratures other than the Lobatto-type Legendre-Gauss quadrature can also be used for integration. In fact, the standard Legendre-Gauss quadrature or the Clenshaw-Curtis quadrature can increase the order of accuracy of the approximate system (2.39) (see [85] for details).

The SE method approximates the operator equation form of (2.39). The extension of the length of time history to an integer multiple $\hat{\tau} = \Gamma T \geq \tau$ of the principal period T with

$$\Gamma = \begin{cases} \text{floor} \left(\frac{\max(\tau, T)}{T} \right) & \text{if } \text{mod}(\max(\tau, T), T) = 0, \\ \text{floor} \left(\frac{\max(\tau, T)}{T} \right) + 1 & \text{otherwise,} \end{cases} \quad (2.44)$$

and the consideration of the residual of (1.5) over one principal period leads to operator equation (1.16), where operator \mathcal{A} is now defined as

$$\mathcal{A}\mathbf{z} = \left\{ \dot{\mathbf{z}}(t) - \mathbf{A}(t)\mathbf{z}(t) - \sum_{p=1}^v \mathbf{B}_p(t)\mathbf{z}(t - \tau_p) : t \in [0, T] \right\}, \quad (2.45)$$

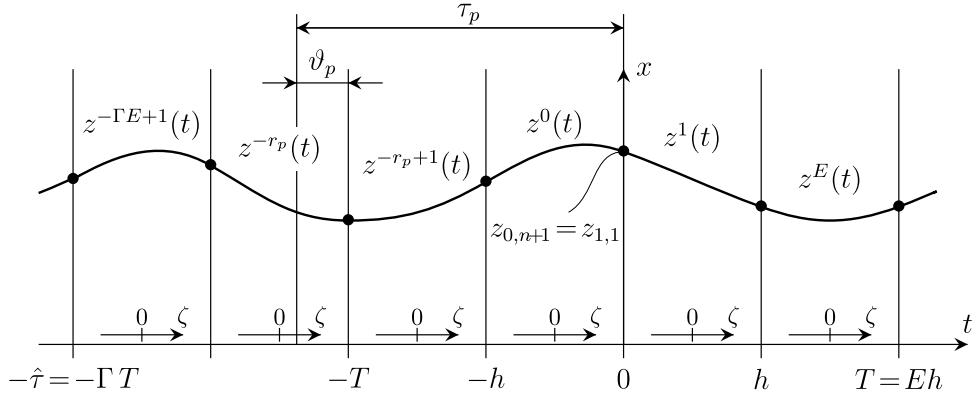


FIGURE 2.4: Splitting of the solution segment \mathbf{z} for the case $s = 1$, $\Gamma = 2$ and $E = 2$. The depicted delay τ_p results in $r_p = 2$.

and the solution segment \mathbf{z} is now given in the form

$$\mathbf{z}(\theta) = \begin{cases} \boldsymbol{\xi}_0(\theta) & \text{if } \theta \in [-\hat{\tau}, 0], \\ \boldsymbol{\xi}(\theta) & \text{if } \theta \in (0, T]. \end{cases} \quad (2.46)$$

The SE method splits this solution segment onto $(\Gamma + 1)E$ number of equidistant sub-segments (referred to as elements in the following) as

$$\mathbf{z}^k = \{\mathbf{z}(\theta) : t \in ((k-1)h, kh]\}, \quad k = -E\Gamma + 1, -E\Gamma + 2, \dots, E; \quad (2.47)$$

where $h = T/E$ denotes the length of elements. The splitting of solution segment \mathbf{z} is illustrated in Figure 2.4 for specific s , Γ and E parameters. Subsequent elements are connected at their boundaries, therefore conditions

$$\mathbf{z}^k(kh) = \mathbf{z}^{k+1}(kh), \quad k = -E\Gamma + 1, -E\Gamma + 2, \dots, E - 1; \quad (2.48)$$

hold. The splitting of solution segment \mathbf{z} transforms (1.16) with (2.45) to a system of operator equations

$$\mathcal{S}^k \mathbf{z}^k - \sum_{p=1}^v \mathcal{Q}^{k,p} \mathbf{z}^{k-r_p-1} - \sum_{p=1}^v \mathcal{R}^{k,p} \mathbf{z}^{k-r_p} = \mathbf{0}, \quad k = 1, 2, \dots, E; \quad (2.49)$$

where the operators are defined as

$$\mathcal{S}^k \mathbf{z}^k = \begin{cases} \dot{\mathbf{z}}^k(t) - \mathbf{A}(t)\mathbf{z}^k(t) & \text{if } t \in ((k-1)h, kh], \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (2.50)$$

$$\mathcal{Q}^{k,p} \mathbf{z}^{k-r_p-1} = \begin{cases} \mathbf{B}_p(t) \mathbf{z}^{k-r_p-1}(t-\tau_p) & \text{if } t \in ((k-1)h, (k-1)h+\vartheta_p], \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (2.51)$$

$$\mathcal{R}^{k,p} \mathbf{z}^{k-r_p} = \begin{cases} \mathbf{B}_p(t) \mathbf{z}^{k-r_p}(t-\tau_p) & \text{if } t \in ((k-1)h + \vartheta_p, kh], \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (2.52)$$

with

$$r_p = \begin{cases} \text{floor}\left(\frac{\tau_p}{h}\right) - 1 & \text{if } \text{mod}(\tau_p, h) = 0, \\ \text{floor}\left(\frac{\tau_p}{h}\right) & \text{otherwise,} \end{cases}, \quad \vartheta_p = \begin{cases} h & \text{if } \text{mod}(\tau_p, h) = 0, \\ \text{mod}(\tau_p, h) & \text{otherwise,} \end{cases} \quad (2.53)$$

(see the illustration in Figure 2.4). By the introduction of element-wise (local) coordinates

$$\zeta^k = \frac{2(t - (k-1)h)}{h} - 1, \quad k = -E\Gamma + 1, -E\Gamma + 2, \dots, E; \quad (2.54)$$

and dropping the index k immediately (see the illustration in Figure 2.4), operators (2.50)–(2.52) assume the form

$$\mathcal{S}^k \mathbf{z}^k = \begin{cases} \frac{2}{h} \frac{d}{d\zeta} \mathbf{z}^k(\zeta) - \mathbf{A} \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \mathbf{z}^k(\zeta) & \text{if } \zeta \in (-1, 1], \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (2.55)$$

$$\mathcal{Q}^{k,p} \mathbf{z}^{k-r_p-1} = \begin{cases} \mathbf{B}_p \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \mathbf{z}^{k-r_p-1}(\zeta + 2 - \hat{\vartheta}_p) & \text{if } \zeta \in (-1, -1 + \hat{\vartheta}_p], \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (2.56)$$

$$\mathcal{R}^{k,p} \mathbf{z}^{k-r_p} = \begin{cases} \mathbf{B}_p \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \mathbf{z}^{k-r_p}(\zeta - \hat{\vartheta}_p) & \text{if } \zeta \in (-1 + \hat{\vartheta}_p, 1], \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (2.57)$$

where $\hat{\vartheta}_p = 2\vartheta_p/h$. The application of element-wise coordinate transformation (2.54) to boundary conditions (2.48) gives

$$\mathbf{z}^k(1) = \mathbf{z}^{k+1}(-1), \quad k = -E\Gamma + 1, -E\Gamma + 2, \dots, E - 1. \quad (2.58)$$

The SE method approximates each element \mathbf{z}^k with its Lagrange interpolant as

$$\tilde{\mathbf{z}}^k(\zeta) = \sum_{j=1}^{n+1} \phi_j(\zeta) \tilde{\mathbf{z}}_{k,j}, \quad k = -E\Gamma + 1, -E\Gamma + 2, \dots, E; \quad (2.59)$$

where $\tilde{\mathbf{z}}_{k,j} = \mathbf{z}^k(\zeta_j)$ and $\phi_j(\zeta)$ are the Lagrange base polynomials. The node set of interpolation $\{\zeta_j\}_{j=1}^{n+1} \subset [-1, 1]$ is chosen to be of Lobatto-type, that is there are nodes on the endpoints of interval $\zeta \in [-1, 1]$, which simplifies boundary conditions (2.58) to

$$\tilde{\mathbf{z}}_{k,n+1} = \tilde{\mathbf{z}}_{k+1,1}, \quad k = -E\Gamma + 1, -E\Gamma + 2, \dots, E - 1 \quad (2.60)$$

(see the illustration in Figure 2.4 for $k = 0$). Note that together with boundary conditions (2.60), (2.59) gives a piecewise Lagrange interpolant for \mathbf{z} . After the substitution of the Lagrange interpolant (2.59) to (2.49), one obtains residual function segments

$$\mathbf{r}^k = \mathcal{S}^k \tilde{\mathbf{z}}^k - \sum_{p=1}^v \mathcal{Q}^{k,p} \tilde{\mathbf{z}}^{k-r_p-1} - \sum_{p=1}^v \mathcal{R}^{k,p} \tilde{\mathbf{z}}^{k-r_p} \neq \mathbf{0}, \quad k = 1, 2, \dots, E. \quad (2.61)$$

In general, the approximate solution $\tilde{\mathbf{z}}^k$ cannot satisfy operator equation (1.16), thus the residual function is nonzero. In order to approximate the monodromy operator, a mapping between coefficients $\{\tilde{\mathbf{z}}_{k,j}\}_{k=1,j=1}^{E,n+1}$ and $\{\tilde{\mathbf{z}}_{k,j}\}_{k=-E\Gamma+1,j=1}^{0,n+1}$ have to be

determined. This requires $sE(n+1)$ number of independent equations. The application of the method of weighted residuals to (2.61) gives

$$\langle \mathbf{r}^k, \psi_i \rangle = \mathbf{0}, \quad k = 1, 2, \dots, E; \quad i = 1, 2, \dots, n+1. \quad (2.62)$$

Here the test functions ψ_i are chosen as the base functions of the subspace \mathcal{P}_n of the approximate solution. The same way as for the PsT method, they are selected to be the Legendre polynomials as $\psi_i = P_{i-1}$. Note that other ways for the selection of test functions also lead to convergent approximation schemes. E.g. in [43] the test function set is determined by the least-square method.

Equations (2.62) do not consider the boundary conditions corresponding to $k = 0, 1, \dots, E-1$ in (2.60). These boundary conditions require the solution to be known, that is they are non-local. In order to enforce the boundary conditions, the tau method is applied. Similarly to the PsT method, equations corresponding to the test function of highest order (that is to P_n) are replaced for each k in (2.62). After this replacement, (2.62) and (2.60) give the system of algebraic equations

$$\tilde{\mathbf{z}}_{1,1} = \tilde{\mathbf{z}}_{0,n+1} \quad (2.63)$$

$$\begin{aligned} & \mathbf{S}_{i,1}^k \tilde{\mathbf{z}}_{k-1,n+1} + \sum_{j=2}^{n+1} \mathbf{S}_{i,j}^k \tilde{\mathbf{z}}_{k,j} - \sum_{p=1}^v \mathbf{Q}_{i,1}^{k,p} \tilde{\mathbf{z}}_{k-r_p-2,n+1} \\ & - \sum_{p=1}^v \sum_{j=2}^{n+1} \mathbf{Q}_{i,j}^{k,p} \tilde{\mathbf{z}}_{k-r_p-1,j} - \sum_{p=1}^v \mathbf{R}_{i,1}^{k,p} \tilde{\mathbf{z}}_{k-r_p-1,n+1} - \sum_{p=1}^v \sum_{j=2}^{n+1} \mathbf{R}_{i,1}^{k,p} \tilde{\mathbf{z}}_{k-r_p,j} = \mathbf{0}, \\ & k = 1, 2, \dots, E; \quad i = 1, 2, \dots, n; \end{aligned} \quad (2.64)$$

where

$$\mathbf{S}_{i,j}^k = \int_{-1}^1 \left(\frac{2}{h} \mathbf{I} \phi'_j(\zeta) - \mathbf{A} \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \phi_j(\zeta) \right) \psi_i(\zeta) d\zeta, \quad (2.65)$$

$$\mathbf{Q}_{i,j}^{k,p} = \int_{-1}^{-1+\hat{\vartheta}_p} \mathbf{B}_p \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \phi_j(\zeta + 2 - \hat{\vartheta}_p) \psi_i(\zeta) d\zeta, \quad (2.66)$$

$$\mathbf{R}_{i,j}^{k,p} = \int_{-1+\hat{\vartheta}_p}^1 \mathbf{B}_p \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \phi_j(\zeta - \hat{\vartheta}_p) \psi_i(\zeta) d\zeta. \quad (2.67)$$

Using Lobatto-type Legendre-Gauss quadrature, the above integrals can be computed numerically as

$$\mathbf{S}_{i,j}^k = \frac{2}{h} \mathbf{I} \sum_{q=1}^{n+1} \sum_{l=1}^{n+1} F_{i,q} L_{q,l} D_{l,j} - \sum_{q=1}^{n+1} F_{i,q} \mathbf{A}_q^k L_{q,j}, \quad (2.68)$$

$$\mathbf{Q}_{i,j}^{k,p} = \sum_{q=1}^{n+1} F_{i,q}^{Q,p} \mathbf{B}_q^{Q,k,p} L_{q,j}^{Q,p}, \quad (2.69)$$

$$\mathbf{R}_{i,j}^{k,p} = \sum_{q=1}^{n+1} F_{i,q}^{R,p} \mathbf{B}_q^{R,k,p} L_{q,j}^{R,p}, \quad (2.70)$$

where terms, corresponding to the periodic coefficients are calculated as

$$\mathbf{A}_q^k = \mathbf{A} \left(\frac{h}{2} (\zeta_q^* + 1) + (k - 1)h \right), \quad (2.71)$$

$$\mathbf{B}_q^{Q,k,p} = \mathbf{B}_p \left(\frac{\vartheta_p}{2} (\zeta_q^* + 1) + (k - 1)h \right), \quad (2.72)$$

$$\mathbf{B}_q^{R,k,p} = \mathbf{B}_p \left(\frac{h - \vartheta_p}{2} (\zeta_q^* + 1) + \vartheta_p + (k - 1)h \right), \quad (2.73)$$

while $F_{i,q}$, $L_{q,j}$ and $D_{k,j}$ are defined in Sections 2.1.1 and 2.1.3 and

$$F_{i,q}^{Q,p} = \frac{w_q \hat{\vartheta}_p}{2} \psi_i \left(\frac{\hat{\vartheta}_p}{2} (\zeta_q^* + 1) - 1 \right), \quad L_{q,j}^{Q,p} = \phi_j \left(\frac{\hat{\vartheta}_p}{2} (\zeta_q^* - 1) + 1 \right), \quad (2.74)$$

$$F_{i,q}^{R,p} = \frac{w_q (2 - \hat{\vartheta}_p)}{2} \psi_i \left(\frac{(2 - \hat{\vartheta}_p) \zeta_q^* + \hat{\vartheta}_p}{2} \right), \quad L_{q,j}^{R,p} = \phi_j \left(\frac{(2 - \hat{\vartheta}_p) \zeta_q^* - \hat{\vartheta}_p}{2} \right). \quad (2.75)$$

Note that if the nodes of interpolation are chosen to be the same as the quadrature nodes, then due to the property (2.9) of the interpolant, (2.68) reduces to

$$\mathbf{S}_{i,j}^k = \frac{2}{h} \mathbf{I} \sum_{q=1}^{n+1} F_{i,q} D_{q,j} - F_{i,j} \mathbf{A}_j^k. \quad (2.76)$$

In this thesis, the nodes of interpolation of the SE method are chosen according to the Lobatto-type Legendre-Gauss quadrature, thus the above simplification of (2.68) applies.

Finally, the nonzero parts of the matrix approximation $\mathbf{U} \in \mathbb{R}^{s(E\Gamma n+1) \times s(E\Gamma n+1)}$ of the monodromy operator are given by

$$\mathbf{U}(1 : sE(\Gamma - 1)n, sEn + 1 : sE\Gamma n) = \mathbf{I}, \quad (2.77)$$

$$\mathbf{U}(sE(\Gamma - 1)n + 1 : s(E\Gamma n + 1)) = \boldsymbol{\Lambda}^{-1} \boldsymbol{\Upsilon}, \quad (2.78)$$

where the elements of matrices $\boldsymbol{\Lambda} \in \mathbb{R}^{s(En+1) \times s(En+1)}$ and $\boldsymbol{\Upsilon} \in \mathbb{R}^{s(En+1) \times s(E\Gamma n+1)}$ are computed according to (2.63)–(2.64) as

$$\boldsymbol{\Lambda}(1 : s, 1 : s) = \mathbf{I}, \quad (2.79)$$

$$\boldsymbol{\Lambda}(s + s(k - 1)n + 1 : s + skn, :) = \boldsymbol{\Lambda}_k, \quad k = 1, 2, \dots, E; \quad (2.80)$$

$$\boldsymbol{\Upsilon}(1 : s, sE\Gamma n + 1 : s + sE\Gamma n) = \mathbf{I}, \quad (2.81)$$

$$\boldsymbol{\Upsilon}(s + s(k - 1)n + 1 : s + skn, :) = \boldsymbol{\Upsilon}_k, \quad k = 1, 2, \dots, E; \quad (2.82)$$

with

$$\boldsymbol{\Lambda}_k = \boldsymbol{\Lambda}_k^S + \sum_{p=1}^v \left(\boldsymbol{\Lambda}_k^{Q,p} + \boldsymbol{\Lambda}_k^{R,p} \right), \quad \boldsymbol{\Upsilon}_k = \sum_{p=1}^v \left(\boldsymbol{\Upsilon}_k^{Q,p} + \boldsymbol{\Upsilon}_k^{R,p} \right), \quad (2.83)$$

and

$$\Lambda_k^S(:, s(k-1)n+1:s+skn) = \left[\mathbf{S}_{i,j}^k \right]_{i=1,j=1}^{n,n+1}, \quad (2.84)$$

$$\Lambda_k^{Q,p}(:, s(k-r_p-2)n+1:s+s(k-r_p-1)n) = \left[-\mathbf{Q}_{i,j}^{k,p} \right]_{i=1,j=1}^{n,n+1}, \text{ if } k \geq r_p+2, \quad (2.85)$$

$$\Lambda_k^{R,p}(:, s(k-r_p-1)n+1:s+s(k-r_p)n) = \left[-\mathbf{R}_{i,j}^{k,p} \right]_{i=1,j=1}^{n,n+1}, \text{ if } k \geq r_p+1, \quad (2.86)$$

$$\Upsilon_k^{Q,p}(:, s(E\Gamma+k-r_p-2)n+1:s+s(E\Gamma+k-r_p-1)n) = \left[\mathbf{Q}_{i,j}^{k,p} \right]_{i=1,j=1}^{n,n+1}, \text{ if } k < r_p+2, \quad (2.87)$$

$$\Upsilon_k^{R,p}(:, s(E\Gamma+k-r_p-1)n+1:s+s(E\Gamma+k-r_p)n) = \left[\mathbf{R}_{i,j}^{k,p} \right]_{i=1,j=1}^{n,n+1}, \text{ if } k < r_p+1. \quad (2.88)$$

The first and second arguments of the above matrices indicate the indices of the rows and the columns, respectively, according to Matlab syntax. Note that in (2.78), Λ and Υ are the finite dimensional approximations of operators \mathcal{A}^+ and \mathcal{A}^- in (1.18), respectively.

The above description provides compact formulas for the construction of the matrix approximation \mathbf{U} of the monodromy operator $\mathcal{U}(T)$ for the general case (2.39). However, application of these formulas to specific examples is not always straightforward. In order to help to understand the process of computation, a computational example is given for particular computational and system parameters for a time-periodic oscillator with two delays.

Time-periodic oscillator with two delays. The first order form of this oscillator reads

$$\dot{\xi}(t) = \mathbf{A}(t)\xi(t) + \mathbf{B}\xi(t-\tau_1) + \mathbf{B}\xi(t-\tau_2), \quad (2.89)$$

where now

$$\mathbf{A}(t) = \begin{bmatrix} 0 & 1 \\ -6 - \varepsilon \cos(2\pi t) & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}. \quad (2.90)$$

Assume that $E = 3$ and $n = 2$, that is 3 elements and 3 interpolation nodes are used. Consequently, the length of the elements is $h = T/3$ and the Lobatto-type Legendre-Gauss quadrature nodes and weights are $\zeta_1^* = -1$, $\zeta_2^* = 0$, $\zeta_3^* = 1$ and $w_1 = 1/3$, $w_2 = 4/3$, $w_3 = 1/3$; respectively (see Appendix A.2). Note that the nodes of interpolation are the same as the quadrature nodes, that is $\zeta_j = \zeta_j^*$ for $j = 1, 2, 3$. The test functions are the Legendre polynomials up to degree 1, hence $\psi_1(\zeta) = 1$ and $\psi_2(\zeta) = \zeta$ (see Appendix A.1). Assume that $2h < \tau_1 < 3h$ and $4h < \tau_2 < 5h$, therefore (2.44) gives $\Gamma = 2$, which results in $r_1 = 2$ and $r_2 = 4$. After $\vartheta_1 = \text{mod}(\tau_1, h)$, $\vartheta_2 = \text{mod}(\tau_2, h)$ and $\hat{\vartheta}_1 = 2\vartheta_1/h$, $\hat{\vartheta}_2 = 2\vartheta_2/h$ are computed, the submatrices (2.68)–(2.70) can be calculated. Matrices $\Lambda \in \mathbb{R}^{14 \times 14}$ and $\Upsilon \in \mathbb{R}^{14 \times 26}$ can be computed according to (2.79)–(2.88). These matrices can be decomposed as

$$\Lambda = \Lambda^0 + \Lambda^S + \sum_{p=1}^2 (\Lambda^{Q,p} + \Lambda^{R,p}), \quad \Upsilon = \Upsilon^0 + \sum_{p=1}^2 (\Upsilon^{Q,p} + \Upsilon^{R,p}), \quad (2.91)$$

where the structure of the matrices are given by

$$\Lambda^S = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ S_{1,1}^1 & S_{1,2}^1 & S_{1,3}^1 & 0 & 0 & 0 & 0 \\ S_{2,1}^1 & S_{2,2}^1 & S_{2,3}^1 & 0 & 0 & 0 & 0 \\ 0 & 0 & S_{1,1}^2 & S_{1,2}^2 & S_{1,3}^2 & 0 & 0 \\ 0 & 0 & S_{2,1}^2 & S_{2,2}^2 & S_{2,3}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & S_{1,1}^3 & S_{1,2}^3 & S_{1,3}^3 \\ 0 & 0 & 0 & 0 & S_{2,1}^3 & S_{2,2}^3 & S_{2,3}^3 \end{bmatrix}, \quad (2.92)$$

$$\Lambda^{R,1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -R_{1,1}^{3,1} & -R_{1,2}^{3,1} & -R_{1,3}^{3,1} & 0 & 0 & 0 & 0 \\ -R_{2,1}^{3,1} & -R_{2,2}^{3,1} & -R_{2,3}^{3,1} & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (2.93)$$

$$\Upsilon^{Q,1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Q_{1,1}^{1,1} & Q_{1,2}^{1,1} & Q_{1,3}^{1,1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Q_{2,1}^{1,1} & Q_{2,2}^{1,1} & Q_{2,3}^{1,1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & Q_{1,1}^{2,1} & Q_{1,2}^{2,1} & Q_{1,3}^{2,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & Q_{2,1}^{2,1} & Q_{2,2}^{2,1} & Q_{2,3}^{2,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & Q_{1,1}^{3,1} & Q_{1,2}^{3,1} & Q_{1,3}^{3,1} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & Q_{2,1}^{3,1} & Q_{2,2}^{3,1} & Q_{2,3}^{3,1} \end{bmatrix}, \quad (2.94)$$

$$\Upsilon^{R,1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & R_{1,1}^{1,1} & R_{1,2}^{1,1} & R_{1,3}^{1,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & R_{2,1}^{1,1} & R_{2,2}^{1,1} & R_{2,3}^{1,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{1,1}^{2,1} & R_{1,2}^{2,1} & R_{1,3}^{2,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & R_{2,1}^{2,1} & R_{2,2}^{2,1} & R_{2,3}^{2,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad (2.95)$$

$$\Upsilon^{Q,2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & Q_{1,1}^{1,2} & Q_{1,2}^{1,2} & Q_{1,3}^{1,2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & Q_{2,1}^{1,2} & Q_{2,2}^{1,2} & Q_{2,3}^{1,2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & Q_{1,1}^{2,2} & Q_{1,2}^{2,2} & Q_{1,3}^{2,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & Q_{2,1}^{2,2} & Q_{2,2}^{2,2} & Q_{2,3}^{2,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & Q_{1,1}^{3,2} & Q_{1,2}^{3,2} & Q_{1,3}^{3,2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & Q_{2,1}^{3,2} & Q_{2,2}^{3,2} & Q_{2,3}^{3,2} & 0 & 0 & 0 \end{bmatrix}, \quad (2.96)$$

$$\Upsilon^{R,2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & R_{1,1}^{1,2} & R_{1,2}^{1,2} & R_{1,3}^{1,2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & R_{2,1}^{1,2} & R_{2,2}^{1,2} & R_{2,3}^{1,2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & R_{1,1}^{2,2} & R_{1,2}^{2,2} & R_{1,3}^{2,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & R_{2,1}^{2,2} & R_{2,2}^{2,2} & R_{2,3}^{2,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & R_{1,1}^{3,2} & R_{1,2}^{3,2} & R_{1,3}^{3,2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & R_{2,1}^{3,2} & R_{2,2}^{3,2} & R_{2,3}^{3,2} & 0 & 0 & 0 \end{bmatrix} \quad (2.97)$$

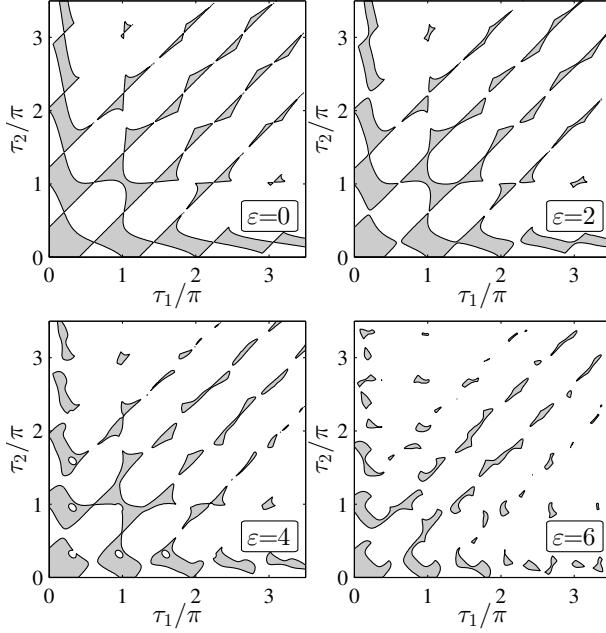


FIGURE 2.5: Stability charts of the time-periodic oscillator with two delays (2.89) for different ε values. The number of elements is $E = 1$, the order of interpolation is $n = 10$.

and

$$\boldsymbol{\Lambda}^0 = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \cdots & \cdots & \mathbf{0} \end{bmatrix}, \quad \boldsymbol{\Upsilon}^0 = \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I} \\ \vdots & & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & & \vdots \\ \mathbf{0} & \cdots & \cdots & \mathbf{0} \end{bmatrix}. \quad (2.98)$$

Furthermore, $\boldsymbol{\Lambda}^{Q,2} = \boldsymbol{\Lambda}^{R,2} = \mathbf{0}$ because $\tau_2 > T$ and $\boldsymbol{\Lambda}^{Q,1} = \mathbf{0}$ because $E < r_1 + 2$. Finally, the matrix approximation $\mathbf{U} \in \mathbb{R}^{26 \times 26}$ of the monodromy operator can be constructed according to (2.77)–(2.78). Note that $\mathbf{B}_q^{Q,k,p} = \mathbf{B}_q^{R,k,p} = \mathbf{B}$ for $p = 1, 2$; $k = 1, 2, 3$ and $q = 1, 2, 3$, hence $\mathbf{Q}_{i,j}^{k,p} = \mathbf{Q}_{i,j}^{k+1,p}$ and $\mathbf{R}_{i,j}^{k,p} = \mathbf{R}_{i,j}^{k+1,p}$ for $k = 1, 2$. In order to determine stability boundaries, the matrix approximation \mathbf{U} of the monodromy operator has to be computed in several points of the (τ_1, τ_2) parameter plane. Note, however, that if τ_1 and τ_2 are changed, then only the terms $F_{i,q}^{Q,p}, F_{i,q}^{R,p}$ and $L_{i,q}^{Q,p}, L_{i,q}^{R,p}$ for $p = 1, 2$; $i = 1, 2$ and $q = 1, 2, 3$ have to be recomputed and all the other terms remain the same. This feature is reflected in the low time-demand of the SE method for the calculation of the stability chart of (2.89), which is shown in Figure 2.5.

2.3 Comparison of spectral methods

In this section the efficiency of the PsT and SE methods are compared with two methods from the literature. Comparison is made based on results, obtained for three linear autonomous DDEs: the Hayes equation, an oscillator with two delays and an oscillator with distributed delay. In order to provide a base for comparison it is necessary to calculate the exact stability boundaries and the rightmost characteristic exponents of the investigated equations. These are detailed in the sequel.

2.3.1 Exact stability boundaries

For the determination of stability boundaries the D-subdivision method [75] is used which utilizes the fact that the system loses stability when a root is crossing the imaginary axis, thus it has the form $\lambda = i\beta$. Substituting this into the characteristic equation (1.13), one can determine co-dimension 1 surfaces in the space of system parameters with running parameter β . When there are only two system parameters these co-dimension 1 surfaces result in the so-called D-curves on the plane of system parameters. These D-curves split the plane of system parameters onto domains where the number of unstable characteristic roots is the same. There are different methods to find the domains with zero unstable roots (i.e., the stable regions). If the number of unstable characteristic roots is known in one domain bounded by the D-curves, then the stable domains can be traced back using the concept of root crossing direction (see Chapter 3.4 in [60]). The number of unstable characteristic roots can also be calculated using Stepan's formulas (see Theorem 2.19. in [75]).

Hayes equation. This equation reads

$$\dot{x}(t) = ax(t) + bx(t - \tau). \quad (2.99)$$

The D-curves are defined by the parametric curves

$$\beta = 0 : b = -a, \quad (2.100)$$

$$\beta\tau \neq k\pi, k \in \mathbb{N} : a = \frac{\beta \cos(\beta\tau)}{\sin(\beta\tau)}, b = \frac{-\beta}{\sin(\beta\tau)} \quad (2.101)$$

in the parameter plane (a, b) with the running parameter $\beta \in [0, \infty)$ (for details, see Chapter 2.1.1 of [29]). For this equation, the case $\tau = 1$ is analyzed throughout this thesis. The D-curves and the domain of stability are shown in Figure 2.6/A. The D-curves are depicted with gray and black colors, and the stable domains are indicated with gray shading and black borders.

Oscillator with two delays. This DDE has the form

$$\ddot{x}(t) + ax(t) = bx(t - \tau_1) + bx(t - \tau_2), \quad (2.102)$$

where $a > 0$ is assumed. The D-curves are given by equations

$$-\left(\frac{2k\pi}{\tau_1 + \tau_2}\right)^2 + a - 2b(-1)^k \cos\left(k\pi\frac{\tau_1 - \tau_2}{\tau_1 + \tau_2}\right) = 0, \quad k \in \mathbb{Z}, \quad (2.103)$$

$$\tau_2 - \tau_1 - \frac{2k + 1}{\sqrt{a}}\pi = 0, \quad k \in \mathbb{Z}. \quad (2.104)$$

Details on derivation can be found at the end of Chapter 3.1 in [75]. Throughout this thesis, parameters $a = 6$ and $b = 1$ are used, thus (2.102) is a special case of (2.89), corresponding to $\varepsilon = 0$. The D-curves and the domain of stability are shown in Figure 2.6/B.

Oscillator with distributed delay. The general form of this DDE is given by

$$\ddot{x}(t) + ax(t) = b \int_{-\tau}^0 \gamma(\theta)x(t + \theta)d\theta, \quad (2.105)$$

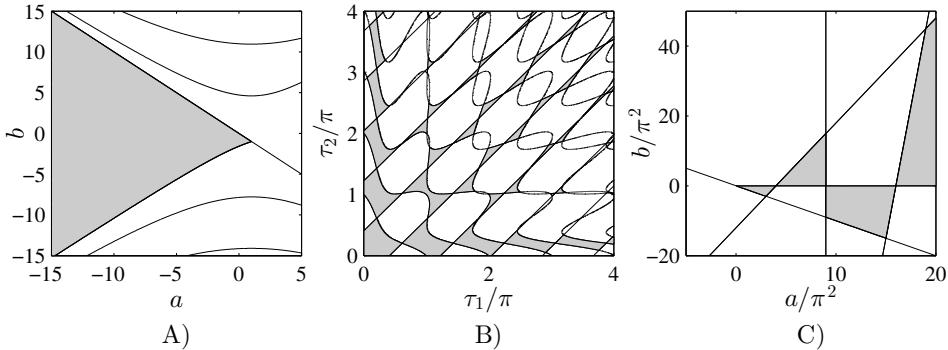


FIGURE 2.6: Exact stability charts with D-curves: A) Hayes equation, B) oscillator with two delays, C) oscillator with distributed delay

where the weight function and the length of time history are chosen as $\gamma(\theta) = \pi \sin(\pi\theta)/2$ and $\tau = 1$, respectively. The D-curves are given by

$$a = (k\pi)^2 - \frac{1 + (-1)^k}{2(1 - k^2)} b, \quad k \in \mathbb{Z} \setminus \{-1, 1\}, \quad (2.106)$$

$$b = 0, \quad k = \pm 1. \quad (2.107)$$

Proof and detailed description on the derivation of stable parameter domains can be found in Theorem 3.26 in [75]. The D-curves and the domain of stability are shown in Figure 2.6/C.

2.3.2 Exact value of the rightmost root

Throughout this chapter, the exact value of the rightmost characteristic root λ_{cr} in any given point of the space of system parameters is determined using the corresponding characteristic equation. After the substitution of $\lambda_{\text{cr}} = \alpha_{\text{cr}} + i\beta_{\text{cr}}$ for the characteristic root, α_{cr} and β_{cr} can be determined by solving the system of non-linear equations given by the real and imaginary parts of the characteristic equation. However, in general, a solution of this system of equations can only be found by using some iterative numerical method. In order to calculate the rightmost root, one should have a close enough initial guess. Here, and in the sequel, this initial guess is produced by the results of the above presented PsT method. In particular, the root used as initial guess is the rightmost root taken from the results of the PsT method after the relative error converged to a value close to machine precision. The numerical solver used for the non-linear characteristic equation is the built-in solver of the software Wolfram Mathematica 9. The Newton-Raphson method was used, such that the precision goal was set to be the double of the precision of the solution based on the PsT method. It has been experienced that starting the iteration from the rightmost root produced by the PsT method, the numerical solution of the characteristic equation gives practically the same result for all the investigated points in the space of system parameters.

2.3.3 Methods under comparison

The PsT and SE methods are compared to two methods from the literature, namely, the pseudospectral collocation (PsC) and the spectral Legendre tau (SLT) method.

In this subsection these methods are briefly described.

Pseudospectral collocation (PsC) method. This method was first proposed by Breda et al [11]. Later, Butcher and Bobrenkov [13] constructed an identical method approaching the problem from the framework of continuous-time approximation [79] and therefore called the technique Chebyshev spectral continuous-time approximation.

Similarly to the PsT method, the PsC method also approximates the OpDE form (1.20) using the method of weighted residuals and enforces the boundary condition using tau approximation. For the PsC method, base functions ϕ_j are Lagrange base polynomials defined by the Lobatto-type Chebyshev nodes, which are given as

$$\zeta_j = \cos\left(\frac{(j-1)\pi}{n}\right), \quad j = 1, \dots, n+1. \quad (2.108)$$

The main difference between the PsC and PsT methods is in the selection of the test function set. The PsT method uses Legendre polynomials as test functions, while the PsC method applies Dirac-delta functions in the form

$$\psi_i(\zeta) = \delta(\zeta - \zeta_i), \quad i = 1, \dots, n; \quad (2.109)$$

where δ denotes the Dirac-delta function and ζ_i are the nodes of interpolation (the Lobatto-type Chebyshev nodes). Note that when using Dirac-delta functions (2.109) as test functions, (2.3) is equivalent to the system of equations obtained by setting the residual function (2.2) to zero at the nodes of interpolation.

The PsC method uses tau approximation and replaces the equation in (2.3) corresponding to $\zeta = 1$ (that is the equation corresponding to $i = n+1$) with the discretized boundary condition (2.7). Consequently, the final form of the approximate system will be (2.17)–(2.20), again. Due to the Dirac-delta test functions, no integration is necessary which simplifies terms in (2.21)–(2.22) by one matrix-matrix multiplication. Furthermore, since the Dirac-delta functions are defined on the nodes of interpolation, terms in (2.21)–(2.22) are simplified by one more matrix-matrix multiplication. Finally, the finite dimensional approximation $\mathbf{G}(t)$ of operator $\mathcal{G}(t)$ is constructed from the sub-matrices

$$\mathbf{G}_{i,j}(t) = \begin{cases} \frac{2}{\tau} \mathbf{I} D_{i,j} & i = 1, \dots, n; \\ \mathcal{L}(t) \phi_j & i = n+1. \end{cases} \quad (2.110)$$

Note that for the calculation of $\mathbf{G}(t)$ no inversion and matrix multiplication are necessary, in contrast with the PsT method.

Spectral Legendre tau (SLT) method. The SLT method was proposed by Vyasarayani et al. [86] for first-order autonomous DDEs with constant delays. Similarly to the PsT and PSC methods, the SLT method also approximates the OpDE form (1.20) using the method of weighted residuals and enforces the boundary condition using tau approximation. The tau approximation is carried out in the same way as for the PsT method, therefore the test functions are $\psi_i = P_{i-1}$, $i = 1, \dots, n$. The, main difference between the PsT and SLT methods is in the base functions. While the PsT method uses Lagrange base polynomials defined by the Chebyshev nodes, the SLT method employs Legendre polynomials as base functions (that is $\phi_j = P_{j-1}$, $j = 1, \dots, n+1$). Consequently, the integral terms in (2.18)–(2.19) can

be calculated as

$$\langle \phi_j, \psi_i \rangle = \langle P_{j-1}, P_{i-1} \rangle, \quad i = 1, \dots, n; \quad (2.111)$$

$$\langle \phi'_j, \psi_i \rangle = \langle P'_{j-1}, P_{i-1} \rangle, \quad i = 1, \dots, n. \quad (2.112)$$

Utilizing the identities and the orthogonality of Legendre polynomials (for details see Appendix A.1), the integral terms in equations (2.111)–(2.112) can be computed in closed form as

$$\langle P_{j-1}, P_{i-1} \rangle = \frac{2}{2j-1} \delta_{i,j}, \quad (2.113)$$

$$\langle P'_{j-1}, P_{i-1} \rangle = \begin{cases} 0 & i > j, \\ 2 & \text{mod}(j-i, 2) \neq 0, \\ 0 & \text{mod}(j-i, 2) = 0, \end{cases} \quad (2.114)$$

where $\delta_{i,j}$ is the Kronecker delta. The final form of the approximate system is given again by (2.17)–(2.20), using the above formulas.

2.3.4 Results of comparison

The results of the comparison of the PsT, PsC, SLT and SE methods are given in Tables 2.1–2.5. The approximate boundaries of stability were determined as follows. Eigenvalues $\tilde{\lambda}$ (or $\tilde{\mu}$) of matrix \mathbf{G} (or \mathbf{U}) were computed for a series of system parameters on an equidistant grid of a particular domain in the plane of system parameters. Eigenvalues, having the largest real part $\tilde{\lambda}_{\text{cr}}$ (or largest absolute value $\tilde{\mu}_{\text{cr}}$) were stored for each gridpoint. A 3-dimensional surface was fitted on these eigenvalues over the parameter plane using the "contour" function of Matlab. The approximate borders of stability were given by the zero-level curve (or level curve at 1) of this 3-dimensional surface. Note that the efficiency of this algorithm can be improved if only the critical eigenvalues are calculated (e.g. see [92]). Efficiency could also be increased if non-uniform grid is used in the parameter plane (for such methods see [9] and [5]).

The primary focus of this thesis is on the stability and stabilizability properties of delayed dynamical systems, therefore the precise calculation of the real part of the rightmost characteristic exponent is of interest. Consequently, the convergence of the approximation of the real part of the rightmost characteristic exponent calculated by different numerical methods was investigated for parameter combinations given in Table 2.1, without the consideration of the error of the imaginary part. Note that corresponding to each characteristic multiplier, only the real part of the characteristic exponent can be determined uniquely as

$$\text{Re}(\lambda) = \frac{1}{h} \ln |\mu|. \quad (2.115)$$

In contrast, infinitely many imaginary parts of the characteristic exponent are introduced for each characteristic multiplier according to

$$\text{Im}(\lambda) = \frac{1}{h} \left(\frac{\text{Im}(\mu)}{\text{Re}(\mu)} \right) + \frac{k\pi}{h}, \quad k = 0, \pm 1, \pm 2, \dots \quad (2.116)$$

Although the imaginary parts of the characteristic exponents do not effect stability,

in physical systems their precise calculation can become important e.g. when calculating the dominant vibration frequencies occurring at the dynamic loss of stability. However, due to the non-uniqueness of the imaginary part (2.116), the reconstruction of dominant vibration frequencies using methods based on the matrix approximation of the monodromy operator (like the SE method) requires additional effort (see e.g. [22]). As a result, when the imaginary parts of the characteristic exponents of autonomous systems are also of interest, methods based on the approximation of the infinitesimal generator (like the PsT, PsC and SLT methods) are preferred.

Table 2.2 presents the stability boundaries for the investigated DDEs using different methods with an increasing order n of approximation. It is observed, that the convergence of stability boundaries is almost exactly the same for the PsT, the SE and the SLT methods, while the PsC method has slower convergence for stability.

2.3 shows the real part of the error $e = \tilde{\lambda}_{\text{cr}} - \lambda_{\text{cr}}$ of the rightmost characteristic exponent as a function of the order of approximation. Results show that the SE method tends to have the highest rate while the PsC method tends to have the lowest rate of convergence. It is interesting to note that the points corresponding to the PsT method and the SLT method coincide for smaller n values. Their convergence rates are close to that of the SE method in case of the Hayes equation and the oscillator with distributed delay, while in case of the oscillator with two delays their convergence rates are close to that of the PsC method. Although Table 2.3 shows the convergence rates only for the parameter combinations given in Table 2.1, the convergence rates were checked by the author in many other points. The selected points show samples for the convergence rates which represent the typical convergence behavior of the analyzed methods. Tables 2.2-2.3 can be used for the comparison of methods based on their convergence with respect to n . However, in practice the time necessary for the computation of the stability charts is also important. Clearly, the time demand of a method depends on its realization as a particular algorithm. A reliable base for comparison could be the number of necessary floating point operations, however, such analysis is not performed in this thesis. Here the computational demand of the methods is characterized by the time necessary for the computation of stability charts.

Table 2.4 shows the computational time of stability charts using 200×200 grid-points on the plane of system parameters. The computational time is presented as a function of approximation order for different equations and different methods. For the calculation of stability charts, the software Matlab was used. During the construction of Matlab codes the author strived to increase the efficiency evenly for algorithms corresponding to the investigated methods in order to provide a base for fair comparison. Based on the structure of the methods under comparison some preliminary estimations can be made on their speed. In all four methods, the majority of computational time is spent on the determination of the eigenvalues of \mathbf{G} (for the PsT, PsC and SLT methods) or \mathbf{U} (for the SE method). Note that the herein presented algorithm calculates all the eigenvalues of these matrices, although for stability only the critical eigenvalue is of interest. It is expected that the structure of the matrix has no significant effect on the time of eigenvalue computation. However, the inversion of matrices \mathbf{A} and \mathbf{N} cost considerable time which has to be done repeatedly in case of the SE method and once in case of the PsT and SLT method. Note also that under updated system parameters the PsC method has to carry out one less matrix-matrix multiplication compared to the other methods. This matrix multiplication however, has small effect on the overall computational time. The results presented in Table 2.4 match with the above discussion, that is the SE method

	Hayes equation		Osc. w. two delays		Osc. w. distr. delay	
	a	b	τ_1/π	τ_2/π	a/π^2	b/π^2
A	-10	5	1.2	0.9	10	-5
B	-5	-10	2.4	1.1	18	18
C	0.5	-1	3	1.5	15	30

TABLE 2.1: Investigated points on the plane of system parameters

requires more computational effort, while the rest of the methods have similar computational demand. In practice, a limit is set for the relative error of the real part of the rightmost exponent and the order of approximation is increased until this limit is reached. Therefore a more useful diagram can be constructed by the combination of Table 2.3 and Table 2.4, which is shown in Table 2.5.

Table 2.5 shows parametric "curves" with the running parameter n on the plane of computational time and error of the real part of the rightmost characteristic exponent. The farthest to the left a curve is placed on this plane the better time efficiency the corresponding method has. Note that while the computational time corresponds to a stability chart, the error corresponds to a particular point of this chart. It is therefore assumed that the time which is necessary for the calculation of the exponents is the same in all points of the parameter plane. The results of Table 2.5 show, that for the Hayes equation the PsT method, for the oscillator with two delays the PsC method, while for the oscillator with distributed delay the SLT method is the most time-efficient. The SE method has the least time-efficiency for the investigated cases. Note, that the SE method can approximate only the real parts of the exponents according to (2.115), therefore, it cannot be used for applications where the location of exponents is of interest on the complex plane (e.g. continuous pole placement [61]). Also note that in case of pseudospectral methods the variables of the discretized system are distinct points of the history function, which can be advantageous in some applications. Furthermore, it should be mentioned that, to the best knowledge of the author, detailed theoretical convergence analysis does not exist for the PsT and SE methods. However, for the PsC and the SLT methods, precise theoretical convergence analysis is provided in [12] and in [33], respectively.

2.4 Extension to hybrid systems

In digitally controlled mechanical systems (see e.g. [76]), the state variables in the governing equations appear with piecewise constant arguments due to sampled data hold in the feedback loop, which results in a hybrid system. In this thesis, the terms with piecewise constant arguments will be referred as terms with *discrete delays*. So far, this thesis have only considered delayed terms whose arguments are continuous functions of time in the analyzed equations. In the following, these terms will be referred as terms with *continuous delays*. Mathematical models involving both discrete and continuous delays often arise in case of haptic devices (see e.g. [40]) where mechanical systems are subjected to both human interaction and digital feedback control. In these models, terms with continuous delay are present due to the reflex delay of the human operator while terms with discrete delay are originated from the sampling and actuation scheme of the digital controller.

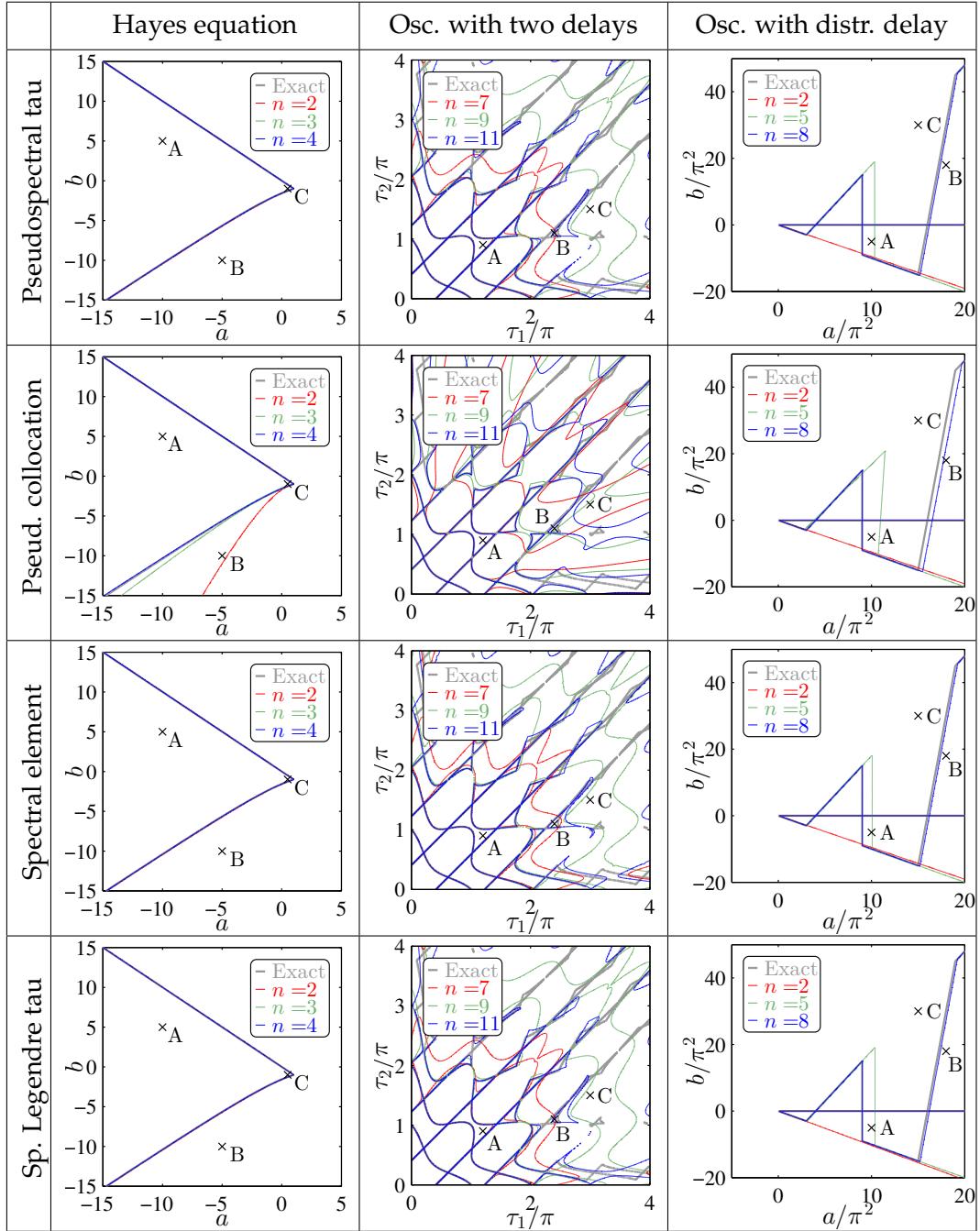


TABLE 2.2: Convergence of stability boundaries using different methods for different DDEs

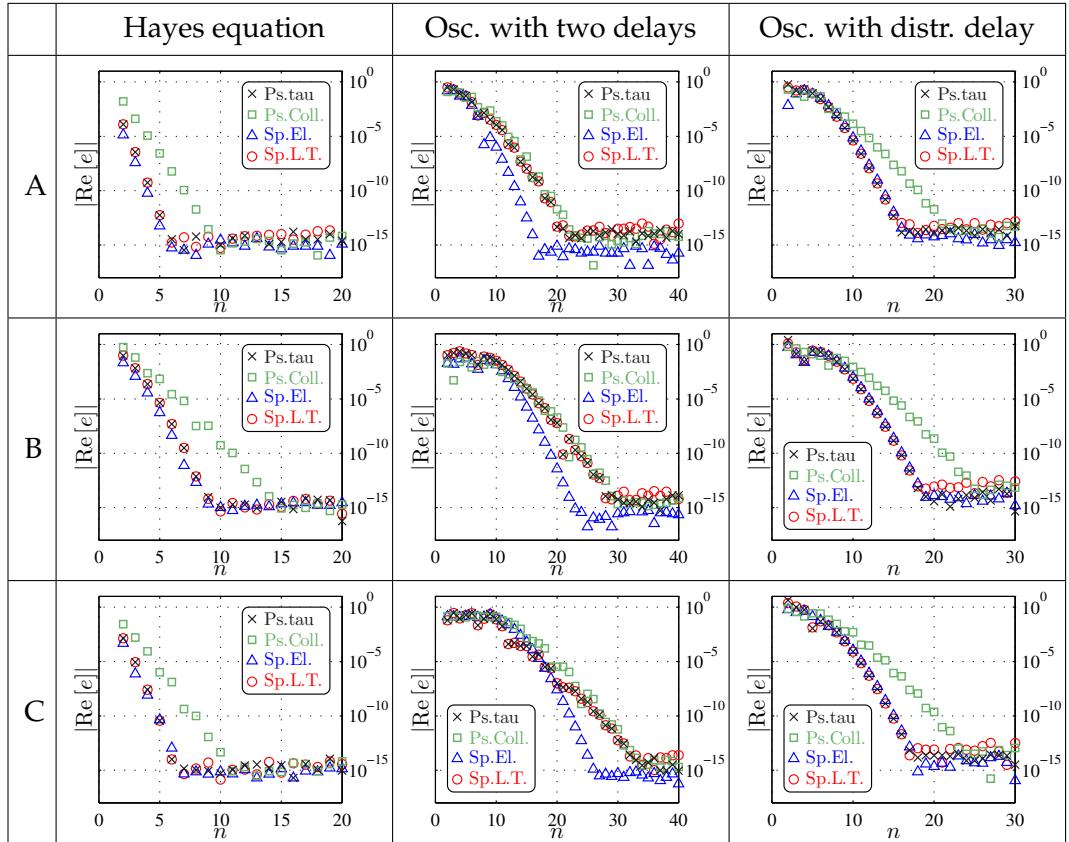


TABLE 2.3: The convergence of the real part of the rightmost exponent using different methods for different DDEs and system parameters according to Table 2.1

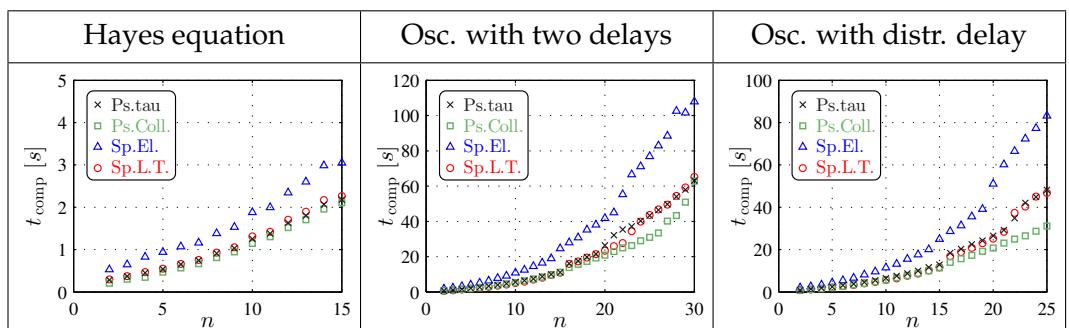


TABLE 2.4: Necessary time for the computation of stability charts based on the calculation of approximate characteristic exponents/multipliers in 200×200 points of the plane of system parameters

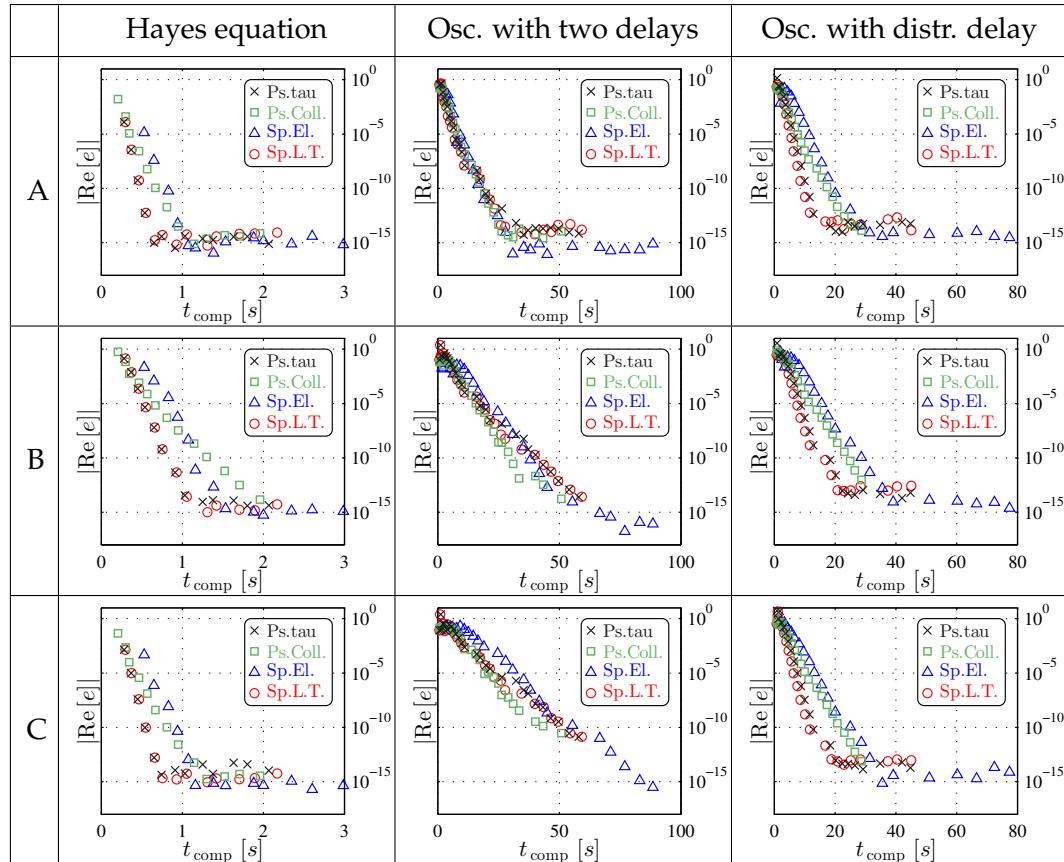


TABLE 2.5: Time efficiency of different methods for different DDEs and system parameters according to Table 2.1

Some further examples are also shown for hybrid time-delay systems in Section 3.2 from the field of machine tool vibrations. In these examples, the feed of the tool is affected by the digital feedback controller of the workpiece-tool system which results in terms with discrete delays, while a term with continuous delay is also present due to the so-called regenerative effect of the cutting process [2, 74].

In spite of the many relevant applications of time-delay systems subjected to digital feedback control, their numerical stability analysis is not well-established in the engineering literature. In the following, the PsT and SE methods are extended for the stability analysis of linear hybrid time-periodic time-delay systems of the form

$$\dot{\xi}(t) = \mathbf{A}(t)\xi(t) + \sum_{p=1}^v \mathbf{B}_p(t)\xi(t - \tau_p) + \mathbf{C}\xi(t_l - \Delta t) + \mathbf{E}\chi_l, \quad t \in [t_l, t_{l+1}), \quad (2.117)$$

$$\chi_l = \chi_{l-1} + \sum_{b=1}^{\tilde{n}} \mathbf{W}_b \xi(t_l - b\Delta t), \quad (2.118)$$

with initial conditions

$$\xi(\theta) = \xi_0(\theta), \quad \theta \in [-\tau, 0], \quad \chi_{-1} = \mathbf{0}. \quad (2.119)$$

In (2.117)–(2.118), a digital feedback controller is added to the open-loop system (2.39). Note that although in (2.117) the terms with continuous delays contain only constant delays (pointwise delays), DDEs involving distributed continuous delay terms can also be approximated by (2.117) using numerical quadratures as detailed in (2.40)–(2.43). The digital control applies state feedback via gain matrix $\mathbf{C} \in \mathbb{R}^{s \times s}$ and the feedback of the integral $\chi_l \in \mathbb{R}^s$ of the state via gain matrix $\mathbf{E} \in \mathbb{R}^{s \times s}$. The state is measured at each sampling instant $t_l + b\Delta t$, $l \in \mathbb{N}$, $b = 1, 2, \dots, \tilde{n}$; while the control input is updated at time instants $t_l = l\Delta T$, $l \in \mathbb{N}$, where Δt is the sampling period and $\Delta T = \tilde{n}\Delta t$ is the actuation period, with $\tilde{n} \in \mathbb{Z}^+$ being the number of samples between two control input updates. The control input is held constant between the two endpoints of each actuation period in accordance to a zero-order hold. In this thesis it is assumed that the calculation of the control input is performed within one sampling period, therefore the state-feedback term in (2.117) has a delay Δt . In (2.118), the integral χ_l is computed numerically using the latest \tilde{n} samples. Matrix \mathbf{W}_b contains the weights of the numerical quadrature used for integration.

In (2.117)–(2.118), $\xi(t - \tau_p)$, $p = 1, 2, \dots, v$ are terms with continuous delays, while $\xi(t_l - b\Delta t)$, $l \in \mathbb{N}$, $b = 1, 2, \dots, \tilde{n}$ are terms with discrete delays. Equations (2.117)–(2.118) form a hybrid system consisting of a time-periodic delay-differential equation (DDE) and a difference equation (DE). In case of a general choice for T and ΔT this system is quasi-periodic. In this thesis it is assumed that $T/\Delta T = \epsilon/\rho$, with $\epsilon, \rho \in \mathbb{Z}^+$, thus $T_p = \epsilon\Delta T = \rho T$ is the principal period of the system. For the case $s = 1$, the sampling and actuation schemes are visualized in Figure 2.7, where the control input function $Q(t)$ is defined as

$$Q(t) = C\xi(t_l - \Delta t) + E\chi_l \quad t \in [t_l, t_{l+1}), \quad (2.120)$$

$$\chi_l = \chi_{l-1} + \sum_{b=1}^{\tilde{n}} W_b \xi(t_l - b\Delta t). \quad (2.121)$$

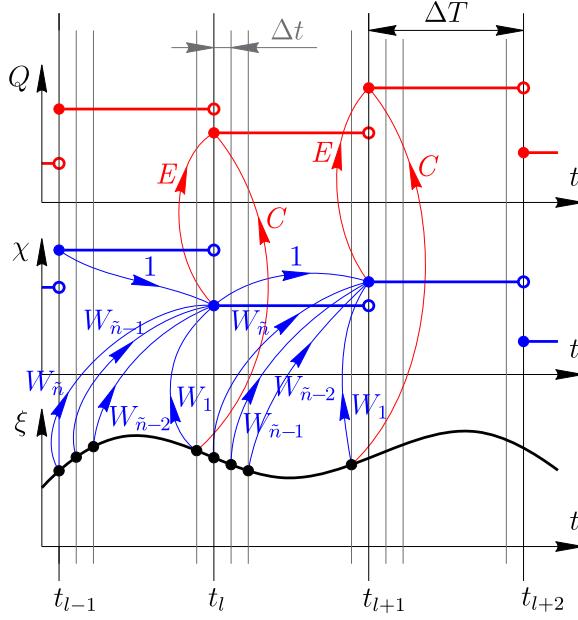


FIGURE 2.7: Visualization of sampling and actuation schemes for the case $s = 1$

The stability of (2.117)–(2.118) is determined by the characteristic multipliers of the monodromy operator $\mathcal{U}(T)$, which is now defined as

$$\boldsymbol{\eta}_{lT_p} = \mathcal{U}(T) \boldsymbol{\eta}_{(l-1)T_p}, \quad l = \mathbb{Z}^+, \quad (2.122)$$

where

$$\boldsymbol{\eta}_{lT_p} = \begin{bmatrix} \boldsymbol{\xi}_{lT_p} \\ \chi_{l\epsilon} \end{bmatrix}. \quad (2.123)$$

In the following, the hybrid system (2.117)–(2.118) is analyzed using the PsT and SE methods.

2.4.1 PsT method

Using function segment $\boldsymbol{\xi}_t$, (2.117) can be converted to the OpDE form

$$\dot{\boldsymbol{\xi}}_t = \mathcal{G}(t)\boldsymbol{\xi}_t + \mathcal{H}\boldsymbol{\xi}_{t_l} + \mathcal{E}\chi_{l-1}, \quad t \in [t_l, t_{l+1}), \quad (2.124)$$

where operators $\mathcal{G}(t)$, \mathcal{H} and \mathcal{E} are defined as

$$\mathcal{G}(t)\boldsymbol{\xi}_t = \begin{cases} \mathbf{A}(t)\boldsymbol{\xi}_t(0) + \sum_{p=1}^v \mathbf{B}_p(t)\boldsymbol{\xi}_t(-\tau_p) & \theta = 0, \\ \frac{d}{d\theta}\boldsymbol{\xi}_t(\theta) & \theta \in [-\tau, 0), \end{cases} \quad (2.125)$$

$$\mathcal{H}\boldsymbol{\xi}_{t_l} = \begin{cases} \mathbf{C}\boldsymbol{\xi}_{t_l}(-\Delta t) + \mathbf{E} \sum_{b=1}^{\tilde{n}} \mathbf{W}_b \boldsymbol{\xi}_{t_l}(-b\Delta t) & \theta = 0, \\ \mathbf{0} & \theta \in [-\tau, 0), \end{cases} \quad (2.126)$$

$$\mathcal{E}\chi_{l-1} = \begin{cases} \mathbf{E}\chi_{l-1} & \theta = 0, \\ \mathbf{0} & \theta \in [-\tau, 0). \end{cases} \quad (2.127)$$

The introduction of function segment ξ_t reformulates (2.118) as

$$\chi_l = \chi_{l-1} + \sum_{b=1}^{\tilde{n}} \mathbf{W}_b \xi_{t_l}(-b\Delta t). \quad (2.128)$$

The discretization of (2.117)–(2.118) is carried out according to the same steps as detailed in Section 2.1. First, the solution segment ξ_t is approximated by its Lagrange interpolant as $\tilde{\xi}_t$ in the form (2.8). Here, the nodes of interpolation are chosen as the Chebyshev points of second kind (2.108). After the substitution of (2.8) to (2.117), the equation is multiplied with test functions $\{\psi_i\}_{i=1}^n$ and integrated over domain $\theta \in [-\tau, 0]$ according to the standard steps of the weighted residual method. The same way as in Section 2.1, test functions are chosen to be the Legendre polynomials. The method of weighted residuals gives n independent equations and the $(n+1)$ -st equation is obtained from the pointwise satisfaction of (2.124) at $\theta = 0$. Consequently, the final form of the approximate system is given by

$$\mathbf{N} \dot{\mathbf{X}}(t) = \mathbf{M}(t) \mathbf{X}(t) + \mathbf{H} \mathbf{X}(t_l) + \mathbf{L} \chi_{l-1}, \quad t \in [t_l, t_{l+1}), \quad (2.129)$$

$$\chi_l = \chi_{l-1} + \mathbf{D} \mathbf{X}(t_l), \quad (2.130)$$

where, after coordinate transformation $\zeta = 2\theta/\tau + 1$, the sub-matrices of \mathbf{N} , $\mathbf{M}(t)$ are given by (2.23)–(2.24), with

$$\mathcal{L}(t) \phi_j = \mathbf{A}(t) \phi_k(1) + \sum_{p=1}^v \mathbf{B}_p(t) \phi_k(1 - 2\tau_p/\tau). \quad (2.131)$$

The sub-matrices of $\mathbf{H} \in \mathbb{R}^{s(n+1) \times s(n+1)}$, $\mathbf{L} \in \mathbb{R}^{s(n+1) \times s}$, $\mathbf{D} \in \mathbb{R}^{s \times s(n+1)}$ and the sub-vectors of $\mathbf{X}(t) \in \mathbb{R}^{s(n+1) \times 1}$ read

$$\mathbf{H}_{i,k} = \begin{cases} \mathbf{0} & i = 1, 2, \dots, n; \\ \mathbf{C} \phi_k(1 - 2\Delta t/\tau) + \mathbf{E} \sum_{b=1}^{\tilde{n}} \mathbf{W}_b \phi_k(1 - 2b\Delta t/\tau) & i = n+1; \end{cases} \quad (2.132)$$

$$\mathbf{L}_i = \begin{cases} \mathbf{0} & i = 1, 2, \dots, n; \\ \mathbf{E} & i = n+1; \end{cases} \quad (2.133)$$

$$\mathbf{D}_k = \sum_{b=1}^{\tilde{n}} \mathbf{W}_b \phi_k(1 - 2b\Delta t/\tau), \quad (2.134)$$

$$\mathbf{X}_k(t) = \tilde{\xi}_t(\tau(\zeta_k - 1)/2). \quad (2.135)$$

Similarly as in (2.1.6), the ODE (2.129) is further approximated using a piecewise constant approximation of $\mathbf{M}(t)$ in order to obtain a matrix approximation for the monodromy operator. In particular, the actuation period ΔT is split onto \tilde{m} intervals and $\mathbf{M}(t)$ is averaged as

$$\tilde{\mathbf{M}}_{l,u} = \frac{1}{\Delta \tilde{t}} \int_0^{\Delta \tilde{t}} \mathbf{M}((l\tilde{m} + u - 1)\Delta \tilde{t} + t) dt, \quad (2.136)$$

where now $\Delta \tilde{t} = \Delta T/\tilde{m}$ with $\tilde{m} \in \mathbb{Z}^+$ and $u = 1, 2, \dots, \tilde{m}; l = 0, 1, \dots, \epsilon - 1$. This way, the time-periodic ODE (2.129) is approximated on $t \in [0, T_p]$ with the series of

autonomous ODEs

$$\mathbf{N}\dot{\tilde{\mathbf{X}}}^{l,u}(t) = \tilde{\mathbf{M}}_{l,u}\tilde{\mathbf{X}}^{l,u}(t) + \mathbf{H}\tilde{\mathbf{X}}^{l,1}(l\tilde{m}\Delta\tilde{t}) + \mathbf{L}\tilde{\chi}_{l-1}, \\ t \in [(l\tilde{m}+u-1)\Delta\tilde{t}, (l\tilde{m}+u)\Delta\tilde{t}], \quad (2.137)$$

$$\tilde{\mathbf{X}}^{l,u}((l\tilde{m}+u-1)\Delta\tilde{t}) = \tilde{\mathbf{X}}^{l,u-1}((l\tilde{m}+u-1)\Delta\tilde{t}), \quad (2.138)$$

and

$$\tilde{\chi}_l = \tilde{\chi}_{l-1} + \mathbf{D}\tilde{\mathbf{X}}^{l,1}(l\tilde{m}\Delta\tilde{t}), \quad (2.139)$$

with $l = 0, \dots, \epsilon - 1$. Here, $\tilde{\mathbf{X}}^{l,u}(t)$ is the approximation of $\mathbf{X}(t)$ in the domain $t \in [(l\tilde{m}+u-1)\Delta\tilde{t}, (l\tilde{m}+u)\Delta\tilde{t}]$. Equations (2.136)–(2.139) result in the discrete mapping

$$\mathbf{Y}_{l+1} = \Phi_l \mathbf{Y}_l, \quad l = 1, 2, \dots, \epsilon - 1; \quad (2.140)$$

between the two endpoints of actuation periods (that is between time instants t_l and t_{l+1}), where

$$\mathbf{Y}_l = \begin{bmatrix} \tilde{\mathbf{X}}^{l,1}(l\tilde{m}\Delta\tilde{t}) \\ \tilde{\chi}_{l-1} \end{bmatrix}, \quad \Phi_l = \begin{bmatrix} \boldsymbol{\alpha}_l & \boldsymbol{\beta}_l \\ \mathbf{D} & \mathbf{I} \end{bmatrix} \quad (2.141)$$

and

$$\boldsymbol{\alpha}_l = \mathbf{J}_l + \mathbf{K}_{l,\tilde{m}} \mathbf{N}^{-1} \mathbf{H}, \quad (2.142)$$

$$\boldsymbol{\beta}_l = \mathbf{K}_{l,\tilde{m}} \mathbf{N}^{-1} \mathbf{L}, \quad (2.143)$$

while

$$\mathbf{J}_l = e^{\tilde{\mathbf{G}}_{l,\tilde{m}}} e^{\tilde{\mathbf{G}}_{l,\tilde{m}-1}} \dots e^{\tilde{\mathbf{G}}_{l,1}}, \quad (2.144)$$

and $\mathbf{K}_{l,\tilde{m}}$ is defined by the recurrence relation

$$\mathbf{K}_{l,u} = e^{\tilde{\mathbf{G}}_{l,u}} \mathbf{K}_{l,u-1} + \Delta\tilde{t} \left(e^{\tilde{\mathbf{G}}_{l,u}} - \mathbf{I} \right) \tilde{\mathbf{G}}_{l,u}^{-1}, \quad (2.145)$$

with

$$\mathbf{K}_{l,0} = \mathbf{0}, \quad \tilde{\mathbf{G}}_{l,u} = \Delta\tilde{t} \mathbf{N}^{-1} \tilde{\mathbf{M}}_{l,u}. \quad (2.146)$$

Finally, the matrix approximation of the monodromy operator is

$$\mathbf{U} = \Phi_{\epsilon-1} \Phi_{\epsilon-2} \dots \Phi_0. \quad (2.147)$$

2.4.2 SE method

Introduce the solution segment

$$\mathbf{z}(\theta) = \begin{cases} \boldsymbol{\xi}_0(\theta) & \text{if } \theta \in [-\hat{\tau}, 0], \\ \boldsymbol{\xi}(\theta) & \text{if } \theta \in (0, T_p], \end{cases} \quad (2.148)$$

and accommodate the definition of Γ in (2.44) as

$$\Gamma = \begin{cases} \text{floor}\left(\frac{\max(\tau, T_p)}{T_p}\right) & \text{if } \text{mod}(\max(\tau, T_p), T_p) = 0, \\ \text{floor}\left(\frac{\max(\tau, T_p)}{T_p}\right) + 1 & \text{otherwise,} \end{cases} \quad (2.149)$$

then the original problem (2.117)–(2.118) can be converted to the operator equation difference equation (OpE-DE) system

$$\mathcal{A}\mathbf{z} - \mathbf{b} = \mathbf{0}, \quad (2.150)$$

$$\boldsymbol{\chi}_{\epsilon-1} = \boldsymbol{\chi}_{-1} + \sum_{a=0}^{\epsilon-1} \sum_{b=1}^{\tilde{n}} \mathbf{W}_b \mathbf{z} (a\Delta T - b\Delta t), \quad (2.151)$$

where operator \mathcal{A} is defined as

$$\mathcal{A}\mathbf{z} = \left\{ \dot{\mathbf{z}}(t) - \mathbf{A}(t)\mathbf{z}(t) - \sum_{p=1}^v \mathbf{B}_p(t)\mathbf{z}(t - \tau_p) - \mathbf{C}\mathbf{z}(t_l - \Delta t) - \mathbf{E} \sum_{a=0}^l \sum_{b=1}^{\tilde{n}} \mathbf{W}_b \mathbf{z}(a\Delta T - b\Delta t) : t \in [t_l, t_{l+1}), l = 0, 1, \dots, \epsilon - 1 \right\} \quad (2.152)$$

and function segment \mathbf{b} is given by

$$\mathbf{b} = \{\mathbf{E}\boldsymbol{\chi}_{-1} : t \in [0, T_p]\}. \quad (2.153)$$

The solution segment \mathbf{z} is then split onto $(\Gamma + 1)E\rho$ number of equidistant subsegments (elements) as

$$\mathbf{z}^k = \{\mathbf{z}(t) : t \in [(k-1)h, kh)\}, \quad k = -\Gamma E\rho + 1, -\Gamma E\rho + 2, \dots, E\rho; \quad (2.154)$$

with $h = T/E$ being the element length. This transforms (2.150)–(2.151) to

$$\begin{aligned} \mathcal{S}^k \mathbf{z}^k - \sum_{p=1}^v \mathcal{Q}^{k,p} \mathbf{z}^{k-r_p-1} - \sum_{p=1}^v \mathcal{R}^{k,p} \mathbf{z}^{k-r_p} \\ - \sum_{l=l_{k-1}}^{l_k} \left(\mathcal{P}^{k,l} \mathbf{z}^{d_{l\tilde{n}-1}} + \sum_{e=1}^{(l+1)\tilde{n}} \mathcal{T}^{k,l,e-\tilde{n}-1} \mathbf{z}^{d_{e-\tilde{n}-1}} \right) - \tilde{\mathbf{b}} \boldsymbol{\chi}_{-1} = \mathbf{0}, \quad k = 1, 2, \dots, E\rho; \end{aligned} \quad (2.155)$$

$$\boldsymbol{\chi}_{\epsilon-1} = \boldsymbol{\chi}_{-1} + \sum_{e=1}^{\epsilon\tilde{n}} \mathbf{W}_{\tilde{n}-\text{mod}(e-1, \tilde{n})} \mathbf{z}^{d_{e-\tilde{n}-1}} ((e-\tilde{n}-1)\Delta t), \quad (2.156)$$

where the operators related to the continuous terms are defined according to (2.50)–(2.52), while operators related to the discrete terms are given as

$$\mathcal{P}^{k,l} \mathbf{z}^{d_{l\tilde{n}-1}} = \begin{cases} \mathbf{C}\mathbf{z}^{d_{l\tilde{n}-1}} (l\Delta T - \Delta t) & \text{if } t \in [\kappa_1^{k,l}, \kappa_2^{k,l}), \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (2.157)$$

$$\begin{aligned} \mathcal{T}^{k,l,e-\tilde{n}-1} \mathbf{z}^{d_{e-\tilde{n}-1}} = \\ \begin{cases} \mathbf{E}\mathbf{W}_{\tilde{n}-\text{mod}(e-1, \tilde{n})} \mathbf{z}^{d_{e-\tilde{n}-1}} ((e-\tilde{n}-1)\Delta t) & \text{if } t \in [\kappa_1^{k,l}, \kappa_2^{k,l}), \\ \mathbf{0} & \text{otherwise,} \end{cases} \end{aligned} \quad (2.158)$$

and \mathbf{b} is divided onto function subsegments as

$$\tilde{\mathbf{b}} = \{\mathbf{E} : t \in [(k-1)h, kh)\}, \quad (2.159)$$

with

$$\kappa_1^{k,l} = \begin{cases} (k-1)h & l = l_{k-1}, \\ (k-1)h + \Delta T(l - l_{k-1}) - \gamma_{k-1} & l_{k-1} < l < l_k, \\ kh - \gamma_k & l = l_k, \end{cases} \quad (2.160)$$

$$\kappa_2^{k,l} = \begin{cases} (k-1)h + \Delta T(l + 1 - l_{k-1}) - \gamma_{k-1} & l_{k-1} \leq l < l_k, \\ kh & l = l_k \end{cases} \quad (2.161)$$

and

$$r_p = \begin{cases} \text{floor}\left(\frac{\tau_p}{h}\right) - 1 & \text{if } \text{mod}(\tau_p, h) = 0, \\ \text{floor}\left(\frac{\tau_p}{h}\right) & \text{otherwise,} \end{cases} \quad \vartheta_p = \begin{cases} h & \text{if } \text{mod}(\tau_p, h) = 0, \\ \text{mod}(\tau_p, h) & \text{otherwise,} \end{cases} \quad (2.162)$$

$$l_k = \text{floor}\left(\frac{kh}{\Delta T}\right), \quad \gamma_k = \text{mod}(kh, \Delta T), \quad d_e = \text{floor}\left(\frac{e\Delta t}{h}\right) + 1. \quad (2.163)$$

The continuity between solution subsegments (2.154) is provided by boundary conditions

$$\mathbf{z}^{k+1}(kh) = \mathbf{z}^k(kh), \quad k = -\Gamma E\rho + 1, -\Gamma E\rho + 2, \dots, E\rho - 1. \quad (2.164)$$

After applying the elementwise coordinate transformation (2.54) and dropping index k immediately (since $\zeta^k \in [-1, 1], \forall k$), equations (2.156)–(2.159) and (2.164) are transformed as follows

$$\chi_{\epsilon-1} = \chi_{-1} + \sum_{e=1}^{\epsilon\tilde{n}} \mathbf{W}_{\tilde{n}-\text{mod}(e-1, \tilde{n})} \mathbf{z}^{d_{e-\tilde{n}-1}} (1 - \tilde{\nu}_{e-\tilde{n}-1}), \quad (2.165)$$

$$\mathcal{P}^{k,l} \mathbf{z}^{d_{l\tilde{n}-1}} = \begin{cases} \mathbf{C} \mathbf{z}^{d_{l\tilde{n}-1}} (1 - \tilde{\nu}_{l\tilde{n}-1}) & \text{if } \zeta \in [\tilde{\kappa}_1^{k,l}, \tilde{\kappa}_2^{k,l}], \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (2.166)$$

$$\begin{aligned} \mathcal{T}^{k,l,e-\tilde{n}-1} \mathbf{z}^{d_{e-\tilde{n}-1}} = & \\ \begin{cases} \mathbf{E} \mathbf{W}_{\tilde{n}-\text{mod}(e-1, \tilde{n})} \mathbf{z}^{d_{e-\tilde{n}-1}} (1 - \tilde{\nu}_{e-\tilde{n}-1}) & \text{if } \zeta \in [\tilde{\kappa}_1^{k,l}, \tilde{\kappa}_2^{k,l}], \\ \mathbf{0} & \text{otherwise,} \end{cases} & \quad (2.167) \end{aligned}$$

$$\tilde{\mathbf{b}} = \{\mathbf{E} : \zeta \in [-1, 1]\}, \quad (2.168)$$

and

$$\mathbf{z}^{k+1}(-1) = \mathbf{z}^k(1), \quad k = -\Gamma E\rho + 1, -\Gamma E\rho + 2, \dots, E\rho - 1; \quad (2.169)$$

respectively, where

$$\tilde{\kappa}_1^{k,l} = \begin{cases} -1 & l = l_{k-1}, \\ -1 + \Delta\tilde{T}(l - l_{k-1}) - \tilde{\gamma}_{k-1} & l_{k-1} < l < l_k, \\ 1 - \tilde{\gamma}_k & l = l_k, \end{cases} \quad (2.170)$$

$$\tilde{\kappa}_2^{k,l} = \begin{cases} -1 + \Delta\tilde{T}(l + 1 - l_{k-1}) - \tilde{\gamma}_{k-1} & l_{k-1} \leq l < l_k, \\ 1 & l = l_k \end{cases} \quad (2.171)$$

and

$$\Delta\tilde{T} = \frac{2\Delta T}{h}, \quad \tilde{\gamma}_k = \frac{2\gamma_k}{h}, \quad \tilde{\vartheta}_p = \frac{2\vartheta_p}{h}, \quad \tilde{\nu}_e = \frac{2(d_e h - e\Delta t)}{h}. \quad (2.172)$$

In each element, the SE method approximates solution subsegments (2.154) with their Lagrange interpolants according to (2.59), where the nodes of interpolation are given by the Lobatto-type Legendre-Gauss quadrature. The system of operator equations (2.155) is then multiplied with test functions $\{\psi_i\}_{i=1}^n$ and integrated over the domain $\zeta \in [-1, 1]$ according to the standard steps of the weighted residual method. For the SE method, the test functions are chosen as the Legendre polynomials up to degree $n - 1$. With the above considerations equations (2.155), (2.165) and (2.169) are discretized as

$$\begin{aligned} \sum_{j=1}^{n+1} \mathbf{S}_{i,j}^k \tilde{\mathbf{z}}_{k,j} &= \sum_{p=1}^v \sum_{j=1}^{n+1} \left(\mathbf{Q}_{i,j}^{k,p} \tilde{\mathbf{z}}_{k-r_p-1,j} + \mathbf{R}_{i,j}^{k,p} \tilde{\mathbf{z}}_{k-r_p,j} \right) \\ &+ \sum_{l=l_{k-1}}^{l_k} \sum_{j=1}^{n+1} \left(\mathbf{P}_{i,j}^{k,l} \tilde{\mathbf{z}}_{d_{l\tilde{n}-1},j} + \sum_{e=1}^{(l+1)\tilde{n}} \mathbf{T}_{i,j}^{k,l,e-\tilde{n}-1} \tilde{\mathbf{z}}_{d_{e-\tilde{n}-1},j} \right) + \tilde{\mathbf{b}}_i \tilde{\chi}_{-1}, \\ i &= 1, 2, \dots, n; \quad k = 1, 2, \dots, E\rho; \end{aligned} \quad (2.173)$$

$$\tilde{\chi}_{\epsilon-1} = \tilde{\chi}_{-1} + \sum_{e=1}^{\epsilon\tilde{n}} \sum_{j=1}^{n+1} \tilde{\mathbf{W}}_{e-\tilde{n}-1,j} \tilde{\mathbf{z}}_{d_{e-\tilde{n}-1},j}, \quad (2.174)$$

and

$$\tilde{\mathbf{z}}_{k+1,1} = \tilde{\mathbf{z}}_{k,n+1}, \quad k = -\Gamma E\rho + 1, -\Gamma E\rho + 2, \dots, E\rho - 1; \quad (2.175)$$

respectively, where the matrices related to continuous terms are given by (2.65)–(2.67) and the matrices related to discrete terms read as

$$\mathbf{P}_{i,j}^{k,l} = \mathbf{C} \phi_j (1 - \tilde{\nu}_{l\tilde{n}-1}) \int_{\tilde{\kappa}_1^{k,l}}^{\tilde{\kappa}_2^{k,l}} \psi_i(\zeta) d\zeta, \quad (2.176)$$

$$\mathbf{T}_{i,j}^{k,l,e-\tilde{n}-1} = \mathbf{E} \tilde{\mathbf{W}}_{e-\tilde{n}-1,j} \int_{\tilde{\kappa}_1^{k,l}}^{\tilde{\kappa}_2^{k,l}} \psi_i(\zeta) d\zeta, \quad (2.177)$$

$$\tilde{\mathbf{b}}_i = \mathbf{E} \int_{-1}^1 \psi_i(\zeta) d\zeta, \quad (2.178)$$

$$\tilde{\mathbf{W}}_{e-\tilde{n}-1,j} = \mathbf{W}_{\tilde{n}-\text{mod}(e-1,\tilde{n})} \phi_j (1 - \tilde{\nu}_{e-\tilde{n}-1}). \quad (2.179)$$

Note that

$$\sum_{l=l_{k-1}}^{l_k} \sum_{e=1}^{(l+1)\tilde{n}} \mathbf{T}_{i,j}^{k,l,e-\tilde{n}-1} = \sum_{e=1}^{(l_{k-1}+1)\tilde{n}} \mathbf{T}_{i,j}^{*,e-\tilde{n}-1} + \sum_{l=l_{k-1}+1}^{l_k} \sum_{e=l\tilde{n}+1}^{(l+1)\tilde{n}} \tilde{\mathbf{T}}_{i,j}^{k,l,e-\tilde{n}-1}, \quad (2.180)$$

where

$$\mathbf{T}_{i,j}^{*,e-\tilde{n}-1} = \mathbf{E} \tilde{\mathbf{W}}_{e-\tilde{n}-1,j} \int_{-1}^1 \psi_i(\zeta) d\zeta, \quad (2.181)$$

$$\tilde{\mathbf{T}}_{i,j}^{k,l,e-\tilde{n}-1} = \mathbf{E} \tilde{\mathbf{W}}_{e-\tilde{n}-1,j} \int_{\tilde{\kappa}_1^{k,l}}^1 \psi_i(\zeta) d\zeta, \quad (2.182)$$

which simplifies the numerical evaluation of the integral term (2.177). The numerical evaluation of the integral terms in (2.65)–(2.67), (2.176), (2.178) and (2.181)–(2.182) give (2.68)–(2.70) for the continuous terms and

$$\mathbf{P}_{i,j}^{k,l} = \mathbf{C} \phi_j(1 - \tilde{\nu}_{l\tilde{n}-1}) \frac{\tilde{\kappa}_2^{k,l} - \tilde{\kappa}_1^{k,l}}{2} \sum_{q=1}^{n+1} \psi_i \left(\frac{\tilde{\kappa}_2^{k,l} - \tilde{\kappa}_1^{k,l}}{2} \zeta_q^* + \frac{\tilde{\kappa}_2^{k,l} + \tilde{\kappa}_1^{k,l}}{2} \right) w_q, \quad (2.183)$$

$$\tilde{\mathbf{b}}_i = \mathbf{E} \sum_{q=1}^{n+1} \psi_i(\zeta_q^*) w_q, \quad (2.184)$$

$$\mathbf{T}_{i,j}^{*,e} = \mathbf{E} \tilde{\mathbf{W}}_{e,j} \sum_{q=1}^{n+1} \psi_i(\zeta_q^*) w_q, \quad (2.185)$$

$$\tilde{\mathbf{T}}_{i,j}^{k,l,e} = \mathbf{E} \tilde{\mathbf{W}}_{e,j} \frac{1 - \tilde{\kappa}_1^{k,l}}{2} \sum_{q=1}^{n+1} \psi_i \left(\frac{1 - \tilde{\kappa}_1^{k,l}}{2} (\zeta_q^* + 1) + \tilde{\kappa}_1^{k,l} \right) w_q, \quad (2.186)$$

for the discrete terms. The non-zero parts of the matrix approximation $\mathbf{U} \in \mathbb{R}^{s(2+E\rho\Gamma n) \times s(2+E\rho\Gamma n)}$ of the monodromy operator are given by

$$\mathbf{U}(1:sE\rho(\Gamma-1)n, sE\rho n+1:sE\rho\Gamma n) = \mathbf{I}, \quad (2.187)$$

$$\mathbf{U}(sE\rho(\Gamma-1)n+1:2s+sE\rho\Gamma n, :) = \mathbf{\Lambda}^{-1} \mathbf{\Upsilon}, \quad (2.188)$$

where the non-zero parts of matrix $\mathbf{\Lambda} \in \mathbb{R}^{s(2+E\rho n) \times s(2+E\rho n)}$ are calculated as

$$\mathbf{\Lambda}(1:s, 1:s) = \mathbf{I}, \quad (2.189)$$

$$\mathbf{\Lambda}(s+s(k-1)n+1:s+skn, 1:s+sE\rho n) = \mathbf{\Lambda}_k, \quad k = 1, 2, \dots, E\rho; \quad (2.190)$$

$$\mathbf{\Lambda}(s+sE\rho n+1:2s+sE\rho n, 1:s+sE\rho n) = \mathbf{\Lambda}^W, \quad (2.191)$$

$$\mathbf{\Lambda}(s+sE\rho n+1:2s+sE\rho n, s+sE\rho n+1:2s+sE\rho n) = \mathbf{I}, \quad (2.192)$$

with

$$\begin{aligned} \mathbf{\Lambda}_k &= \mathbf{\Lambda}_k^S + \sum_{p=1}^v \left(\mathbf{\Lambda}_k^{Q,p} + \mathbf{\Lambda}_k^{R,p} \right) \\ &\quad + \sum_{l=l_{k-1}}^{l_k} \mathbf{\Lambda}_{k,l}^P + \sum_{e=1}^{(l_{k-1}+1)\tilde{n}} \mathbf{\Lambda}_{e-\tilde{n}-1}^{T^*} + \sum_{l=l_{k-1}+1}^{l_k} \sum_{e=l\tilde{n}+1}^{(l+1)\tilde{n}} \mathbf{\Lambda}_{k,l,e-\tilde{n}-1}^T, \end{aligned} \quad (2.193)$$

$$\boldsymbol{\Lambda}^W = \sum_{e=1}^{\epsilon\tilde{n}} \boldsymbol{\Lambda}_{e-\tilde{n}-1}^W, \quad (2.194)$$

where $\boldsymbol{\Lambda}_k^S$, $\boldsymbol{\Lambda}_k^{Q,p}$, $\boldsymbol{\Lambda}_k^{R,p}$ are computed according to (2.84)–(2.86) and

$$\boldsymbol{\Lambda}_{k,l}^P(:, s(d_{l\tilde{n}-1}-1)n+1:s+sd_{l\tilde{n}-1}n) = \left[-\mathbf{P}_{i,j}^{k,l} \right]_{i=1,j=1}^{n,n+1}, \text{ if } d_{l\tilde{n}-1} \geq 1, \quad (2.195)$$

$$\boldsymbol{\Lambda}_e^{T^*}(:, s(d_e-1)n+1:s+sd_en) = \left[-\mathbf{T}_{i,j}^{*,e} \right]_{i=1,j=1}^{n,n+1}, \text{ if } d_e \geq 1, \quad (2.196)$$

$$\boldsymbol{\Lambda}_{k,l,e}^{\tilde{T}}(:, s(d_e-1)n+1:s+sd_en) = \left[-\tilde{\mathbf{T}}_{i,j}^{k,l,e} \right]_{i=1,j=1}^{n,n+1}, \text{ if } d_e \geq 1, \quad (2.197)$$

$$\boldsymbol{\Lambda}_e^W(:, s(d_e-1)n+1:s+sd_en) = \left[-\tilde{\mathbf{W}}_{e,j} \right]_{j=1}^{n+1}, \text{ if } d_e \geq 1. \quad (2.198)$$

The non-zero parts of matrix $\boldsymbol{\Upsilon} \in \mathbb{R}^{s(2+nE\rho) \times s(2+nE\rho\Gamma)}$ are calculated as

$$\boldsymbol{\Upsilon}(1:s, sE\rho\Gamma n+1:s+sE\rho\Gamma n) = \mathbf{I}, \quad (2.199)$$

$$\boldsymbol{\Upsilon}(s+s(k-1)n+1:s+skn, 1:s+sE\rho\Gamma n) = \boldsymbol{\Upsilon}_k, \quad k = 1, 2, \dots, E\rho; \quad (2.200)$$

$$\boldsymbol{\Upsilon}(s+s(k-1)n+1:s+skn, s+sE\rho\Gamma n+1:2s+sE\rho\Gamma n) = \left[\tilde{\mathbf{b}}_i \right]_{i=1}^{n+1}, \quad k = 1, 2, \dots, E\rho; \quad (2.201)$$

$$\boldsymbol{\Upsilon}(s+sE\rho n+1:2s+sE\rho n, 1:s+sE\rho\Gamma n) = \boldsymbol{\Upsilon}^W, \quad (2.202)$$

$$\boldsymbol{\Upsilon}(s+sE\rho n+1:2s+sE\rho n, s+sE\rho\Gamma n+1:2s+sE\rho\Gamma n) = \mathbf{I}, \quad (2.203)$$

with

$$\begin{aligned} \boldsymbol{\Upsilon}_k = & \sum_{p=1}^v \left(\boldsymbol{\Upsilon}_k^{Q,p} + \boldsymbol{\Upsilon}_k^{R,p} \right) \\ & + \sum_{l=l_{k-1}}^{l_k} \boldsymbol{\Upsilon}_{k,l}^P + \sum_{e=1}^{(l_{k-1}+1)\tilde{n}} \boldsymbol{\Upsilon}_{e-\tilde{n}-1}^{T^*} + \sum_{l=l_{k-1}+1}^{l_k} \sum_{e=l\tilde{n}+1}^{(l+1)\tilde{n}} \boldsymbol{\Upsilon}_{k,l,e-\tilde{n}-1}^{\tilde{T}}, \end{aligned} \quad (2.204)$$

$$\boldsymbol{\Upsilon}^W = \sum_{e=1}^{\epsilon\tilde{n}} \boldsymbol{\Upsilon}_{e-\tilde{n}-1}^W, \quad (2.205)$$

where $\boldsymbol{\Upsilon}_k^{Q,p}$, $\boldsymbol{\Upsilon}_k^{R,p}$ are calculated according to (2.87)–(2.88) and

$$\boldsymbol{\Upsilon}_{k,l}^P(:, s(E\rho\Gamma+d_{l\tilde{n}-1}-1)n+1:s+s(E\rho\Gamma+d_{l\tilde{n}-1})n) = \left[\mathbf{P}_{i,j}^{k,l} \right]_{i=1,j=1}^{n,n+1}, \text{ if } d_{l\tilde{n}-1} < 1, \quad (2.206)$$

$$\boldsymbol{\Upsilon}_e^{T^*}(:, s(E\rho\Gamma+d_e-1)n+1:s+s(E\rho\Gamma+d_e)n) = \left[\mathbf{T}_{i,j}^{*,e} \right]_{i=1,j=1}^{n,n+1}, \text{ if } d_e < 1, \quad (2.207)$$

$$\boldsymbol{\Upsilon}_{k,l,e}^{\tilde{T}}(:, s(E\rho\Gamma+d_e-1)n+1:s+s(E\rho\Gamma+d_e)n) = \left[\tilde{\mathbf{T}}_{i,j}^{k,l,e} \right]_{i=1,j=1}^{n,n+1}, \text{ if } d_e < 1, \quad (2.208)$$

$$\boldsymbol{\Upsilon}_e^W(:, s(E\rho\Gamma+d_e-1)n+1:s+s(E\rho\Gamma+d_e)n) = \left[\tilde{\mathbf{W}}_{e,j} \right]_{j=1}^{n+1}, \text{ if } d_e < 1. \quad (2.209)$$

In summary, the steps of the computation of \mathbf{U} are the following. First, the parameters in (2.149), (2.162)–(2.163) and (2.170)–(2.172) are calculated, then the terms in (2.68)–(2.70) and (2.183)–(2.186) are computed for all indices. Thereafter matrices $\boldsymbol{\Lambda}$

and Υ are assembled according to (2.189)–(2.198) and (2.199)–(2.209), respectively. Finally, the matrix approximation \mathbf{U} of the monodromy operator is constructed as (2.187)–(2.188).

2.5 New results

I have introduced the pseudospectral tau method for the finite-dimensional approximation of linear delay-differential equations. The results are summarized as follows.

Thesis 1

The pseudospectral tau (PsT) method can be used for the finite-dimensional approximation of delay-differential equations of the form

$$\dot{\xi}(t) = \mathbf{A}(t)\xi(t) + \sum_{p=1}^v \mathbf{B}_p(t)\xi(t-\tau_p(t)) + \sum_{b=1}^m \int_{-\sigma_{b-1}}^{-\sigma_b} \gamma_b(t, \theta) \xi(t+\theta) d\theta,$$

with a system of ordinary differential equations. Based on the results of numerical experiments, the following statements can be made.

- 1) *The stability properties of the approximating system of ordinary differential equations converge to the stability properties of the above equation with respect to the increase of polynomial order: the stability boundaries converge to the exact ones for both autonomous and non-autonomous systems; the real part of the rightmost root converges for autonomous systems.*
- 2) *For the Hayes equation and for the oscillator with distributed delay, the convergence rate of the PsT method with respect to the increase of the polynomial degree shows the same convergence order as the spectral element (SE) and the spectral Legendre-tau (SLT) methods, while it has better convergence order than that of the pseudospectral collocation (PsC) method. For the oscillator with two delays, the SE has better convergence order than that of the PsC, SLT and PsT methods.*
- 3) *Considering the computation time of the stability diagrams of the Hayes equation, oscillator with two delays, and oscillator with distributed delay, the time-need of the PsT method is less than that of the SE method, same as that of the SLT method and higher than that of the PsC method.*

Related publications: [26, 44, 50]

I have generalized the spectral element method for linear systems with time-periodic coefficients and distributed delays and derived explicit formulas for the matrix approximation of the monodromy operator. The results are summarized as follows.

Thesis 2

The spectral element method can be extended for the stability analysis of time-periodic delay-differential equations of the form

$$\dot{\xi}(t) = \mathbf{A}(t)\xi(t) + \sum_{p=1}^v \mathbf{B}_p(t)\xi(t-\tau_p) + \sum_{b=1}^m \int_{-\sigma_{b-1}}^{-\sigma_b} \gamma_b(t, \theta) \xi(t+\theta) d\theta.$$

Explicit formulas can be determined for the calculation of the matrix approximation \mathbf{U} of the monodromy operator of this dynamical system. Using these formulas, \mathbf{U} can be composed for arbitrary time-periodic coefficient matrices $\mathbf{A}(t)$, $\mathbf{B}_p(t)$, $\gamma_b(t, \theta)$. According to numerical experiments, the stability boundaries determined by the method converge with respect to the increase of the polynomial order.

Related publications: [43, 48]

I have extended both the pseudospectral tau and the spectral element methods for the stability analysis of linear time-periodic hybrid systems with both continuous-time delay (point delay) and discrete-time delay (terms with piecewise constant arguments). The results are summarized as follows.

Thesis 3

The pseudospectral tau and the spectral element methods can be applied for the calculation of the matrix approximation of the monodromy operator of time-periodic hybrid time delay systems of the form

$$\begin{aligned}\dot{\xi}(t) &= \mathbf{A}(t)\xi(t) + \sum_{p=1}^v \mathbf{B}_p(t)\xi(t-\tau_p) + \mathbf{C}\xi(t_l - \Delta t) + \mathbf{E}\chi_l, \quad t \in [t_l, t_{l+1}), \\ \chi_l &= \chi_{l-1} + \sum_{q=1}^{\tilde{n}} \mathbf{W}_b \xi(t_l - b\Delta t),\end{aligned}$$

where $t_l = l\tilde{n}\Delta t$ and $l \in \mathbb{N}$. Based on numerical experiments, both methods converge to the same stability boundaries with respect to the increase of polynomial order.

Related publications: [\[49\]](#)

Chapter 3

Applications to machine tool chatter

Machine tool chatter is the large amplitude vibration between the tool and the workpiece involving intermittent loss of contact. These large amplitude vibrations are harmful to the machining process since they increase the tool wear, lead to poor surface quality or even damage the tool. Since the pioneering work of Tobias and Tlusty [83, 84], the most commonly accepted explanation for chatter is the so-called regenerative effect [2, 69, 74]. In fact, in many publications machine tool chatter is referred to as regenerative chatter. The main point of the regenerative phenomenon is that the cutting force is determined by both the current and the delayed positions of the tool. For turning processes, the tool cuts the surface which was modulated by the tool one revolution earlier. For milling processes, the cutting edge cuts the surface which was modulated by the previous cutting edge one tooth-pass period earlier. In the next tooth-pass these vibration marks generate further undulations of the tool–workpiece system. Due to the hereditary nature of the regenerative phenomenon, the equations of motion of the machine-tool-workpiece system are governed by a system of delay-differential equations (DDEs).

In Computer Numerical Control (CNC) systems machining parameters are usually selected rather conservatively in order to avoid chatter and its detrimental effects. Consequently, CNC systems often use sub-optimal machining parameters which results in decreased productivity. This limitation in machining systems has led to a large body of research on increasing productivity while avoiding chatter vibrations. As a result, chatter prediction, avoidance and control has become an important research field and the last decade has seen a steady increase in the number of the corresponding publications [69]. Many of these publications are concerned with the identification of chatter-free regimes in the process parameter space. One tool that has been extensively used to illustrate these regimes is the stability lobe diagram (SLD). SLDs chart the domains of technological parameters where the stationary cutting operation remains stable for small perturbations. In order to increase the reliability of the SLDs and to account for cutting process uncertainties and parameter shifts, several SLDs may need to be reconstructed at different stages of the cutting process. Furthermore, SLDs need to be recalculated when passive [59, 73, 91] or active [17, 65] chatter control strategies are applied. The latter is necessitated by the repeated tuning of the control parameters which requires fast computation of the SLDs. Consequently, it is important to seek fast algorithms for the calculation of SLDs. Some of the existing numerical and semi-analytical methods for the stability analysis of machining operations include the multi-frequency solution [3, 6], semi-discretization [29, 30], full-discretization [18, 20, 55, 70], numerical integration [19], Runge-Kutta methods [66] and numerical simulation [94]. In many of these methods, the primary concern has been the accuracy and the range

of application of the approach.

Note that the spectral element (SE) method approximates an infinite dimensional difference equation (DE) ((1.3) evaluated at $t = T$), while the methods compared with the SE method in Section 2.3 approximate the infinite dimensional differential equation (1.20) with their finite dimensional approximations (2.5). For autonomous DDEs, Table 2.5 shows that those approximating (1.20), in most cases, have better efficiency than the SE method. On the other hand, for time-periodic DDEs those methods which are based on (1.20), have to apply additional discretization steps in order to express the monodromy operator (as it was shown for the pseudospectral tau (PsT) method in Section 2.1.6). This results in the increased computational need of the methods based on (1.20). In particular, the SE method always obtains the monodromy operator in a single step, while e.g. for (2.36) the PsT method needs $\tilde{m} = 20$ steps in order to have accurate results. Motivated by the need for taking the speed of SLD computations into consideration, this chapter presents an efficient application of the SE method to the calculation of SLDs of milling processes. Since milling processes are described by time-periodic DDEs, the application of methods based on (1.20) are not detailed here because they have inferior convergence properties. A comparison between the SE method and the most widespread time-domain methods of the machining literature is carried out based on the efficiency of SLD calculations.

Without the consideration of the regenerative phenomenon, the available maximum depth of cut and maximum spindle speed depends mainly on the limitations of the machine tool structure and the spindle unit of the machining center. However, the occurrence of regenerative chatter gives an additional limitation (visualized by SLDs) to the maximum depth of cut and maximum spindle speed. Consequently, even with the accurate identification of SLDs, chatter still gives a limitation to the full utilization of the capacity of machining centers. As a result, it has been a subject of great interest in the machining literature how the stable domains in SLDs can be increased (thus chatter can be suppressed). There exist passive [73, 82], semi-active [59, 93] and active [17, 65, 72] methods for the suppression of machine tool chatter.

Active chatter suppression techniques aim to reduce chatter vibrations by building an additional controller or control loop to the spindle or to the tool. Practical realizations of the control system, however, involve implementation imperfections. In case of digital control, the sampling effect attributes an intermittent dynamics to the otherwise continuous-time system. This results in a hybrid system involving terms with continuous delay and terms with discrete delay in the governing equations [47, 52]. Also, the control loop involves feedback delays which may interfere with the regenerative delay of the system. Note that for high-speed machining operations, the tooth passing frequency and the natural frequencies of the spindle are in the region of the sampling frequency of the controller. Consequently, the modeling of the sampling effect and the delay of the feedback loop can have importance. However, apart from some studies [47, 53], most existing models of active chatter suppression neglect the sampling effect and the delay of the feedback loop. Such study has not yet been presented for milling, mostly due to the required high computational effort of the stability analysis based on the existing standard methods of the literature. In this chapter the PsT and SE methods are applied for the stability analysis of the mathematical models of machining processes subjected to digital position control.

3.1 Milling operations

This subsection presents the extension of the SE method and its application to the stability analysis of the most common models of milling processes. Furthermore, a comparison of the SE method with the most widespread time-domain methods of the engineering literature is presented in terms of the efficiency of SLD computations.

3.1.1 Milling models

First, the most common mathematical models of machine tool chatter in milling operations are briefly reviewed. Throughout this thesis, milling is assumed with straight-fluted tools and uniformly-distributed cutting teeth and the mechanical models utilize the circular tooth path approximation. The variational systems of the stationary motions of single- and two-degree-of-freedom (DoF) models are presented. These models are often applied in the machining literature [18–21, 32, 55, 66, 70]. The derivation of the below given equations is not described here but can be found in Chapters 5.2.1 and 5.2.4 of [29].

In both the single and two DoF cases, equations determining the stability of the stationary motion are given in the form

$$\dot{\xi}(t) = \mathbf{A}_0 \xi(t) - \mathbf{B}(t) \xi(t) + \mathbf{B}(t) \xi(t - \tau), \quad (3.1)$$

where $\xi \in \mathbb{R}^s$, with $s/2$ being the DoF of the model, $\mathbf{B}(t) = \mathbf{B}(t + \tau)$ for all t time instances. In the case of a tool with uniformly distributed cutting teeth, the principal period is the tooth passing period $\tau = 60/(\Omega v)$, and matrix $\mathbf{B}(t)$ is constructed as

$$\mathbf{B}(t) = \sum_{p=1}^v \mathbf{B}_p(t). \quad (3.2)$$

The spindle speed of the tool, measured in rpm, is denoted by Ω and the number of cutting teeth is given by v . For the single and two DoF models, state vector $\xi(t)$ and matrices \mathbf{A}_0 and $\mathbf{B}_p(t)$ are different, as will be shown in the following.

One DoF model. For the single DoF model, the state vector and the system matrices are

$$\xi(t) = \begin{bmatrix} \xi(t) \\ \dot{\xi}(t) \end{bmatrix}, \quad \mathbf{A}_0 = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix}, \quad \mathbf{B}_p(t) = \tilde{w} \begin{bmatrix} 0 & 0 \\ H_p(t) & 0 \end{bmatrix}, \quad (3.3)$$

where ζ is the damping ratio (with the assumption of proportional damping) and ω_n is the natural angular frequency of the system, while $\tilde{w} = w q_c f_N^{q_c-1} K_n / m_t$ denotes the specific cutting force coefficient, with w being the depth of cut, m_t being the modal mass of the tool, K_n being the normal cutting force coefficient, q_c being the cutting force exponent and $f_N = v_f \tau$ being the nominal feed rate. The feed velocity is denoted by v_f . Note that q_c cutting force exponent assumes that the cutting forces are calculated according to a power law (for more details see [39] or (3.32)–(3.33)). However, it is important to mention that cutting force characteristic other than power law also exist in the literature [77]. State variable $\xi(t)$ is a perturbation around the periodic stationary motion $\bar{x}(t)$ of the tool relative to the workpiece, thus the motion of the tool is described by $x(t) = \bar{x}(t) + \xi(t)$. Here, we assume that the vibrations are parallel to the feed and x gives the position of the tip of the tool.

The dimensionless specific cutting force coefficient corresponding to tooth p is

$$H_p(t) = g_p(t) \sin^{q_c}(\varphi_p(t)) (K_r \cos(\varphi_p(t)) + \sin(\varphi_p(t))), \quad (3.4)$$

which is a periodic function with principal period τ . The cutting force coefficient ratio is $K_r = K_t/K_n$, where K_t denotes the tangential cutting force coefficient. In (3.4), the functions

$$\varphi_p(t) = (2\pi\Omega/60)t + (p-1)2\pi/v, \quad p = 1, 2, \dots, v \quad (3.5)$$

give the angular displacements of the cutting teeth for the case of uniformly distributed cutting teeth and the window function

$$g_p(t) = \begin{cases} 1 & \text{if } \varphi_{\text{ent}} \leq \text{mod}(\varphi_p(t), 2\pi) \leq \varphi_{\text{ex}}, \\ 0 & \text{otherwise,} \end{cases} \quad (3.6)$$

determines whether the p^{th} tool is in or out of the cut, respectively. For up-milling, the tool's angle of entrance is $\varphi_{\text{ent}} = 0$ and its angle of exit is $\varphi_{\text{ex}} = \arccos(1 - 2a_e/D)$, where a_e is the radial immersion, D is the diameter of the tool, and their ratio a_e/D is the radial immersion ratio. For down-milling, $\varphi_{\text{ent}} = \arccos(2a_e/D - 1)$ and $\varphi_{\text{ex}} = \pi$.

Note that when $q_c = 1$ and $H(t) = \sum_{p=1}^v H_p(t) \equiv 1$, equation (3.1) with (3.3) gives the variational system of the standard single DoF model of turning processes (see Chapter 5.1.2 of [29]). This occurs for full immersion ($a_e/D = 1$) milling when the number of teeth is $v = 4$. Consequently, with respect to linear stability analysis, the standard model of turning processes is incorporated in (3.1) and (3.3).

Two DoF model. In case of the two DoF model, the state vector and the system matrices are

$$\xi(t) = \begin{bmatrix} \xi(t) \\ \eta(t) \\ \dot{\xi}(t) \\ \dot{\eta}(t) \end{bmatrix}, \quad \mathbf{A}_0 = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{K} & -\mathbf{C} \end{bmatrix}, \quad \mathbf{B}_p(t) = \tilde{w} \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{H}_p(t) & \mathbf{0} \end{bmatrix}, \quad (3.7)$$

where

$$\mathbf{K} = \omega_n^2 \mathbf{I}, \quad \mathbf{C} = 2\zeta\omega_n \mathbf{I}, \quad \mathbf{H}_p(t) = g_p(t) \sin^{q_c-1}(\varphi_p(t)) \mathbf{T}_p(t) \mathbf{K}_c \mathbf{Q}_p^T(t), \quad (3.8)$$

with \mathbf{I} being a 2×2 identity matrix and

$$\mathbf{T}_p(t) = \begin{bmatrix} \cos(\varphi_p(t)) & \sin(\varphi_p(t)) \\ -\sin(\varphi_p(t)) & \cos(\varphi_p(t)) \end{bmatrix}, \quad \mathbf{K}_c = \begin{bmatrix} K_r \\ 1 \end{bmatrix}, \quad \mathbf{Q}_p(t) = \begin{bmatrix} \sin(\varphi_p(t)) \\ \cos(\varphi_p(t)) \end{bmatrix}. \quad (3.9)$$

The motion of the tool is described by the vector

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} \bar{x}(t) \\ \bar{y}(t) \end{bmatrix} + \begin{bmatrix} \xi(t) \\ \eta(t) \end{bmatrix} \quad (3.10)$$

where $\xi(t)$ and $\eta(t)$ define a perturbation around the stationary motion given by $\bar{x}(t)$ and $\bar{y}(t)$. Here $x(t)$ and $y(t)$ describe the tool's motion parallel to and perpendicular to the feed velocity, respectively. It is assumed that the modal parameters are the same in x and y directions.

3.1.2 Application of the SE method

Several papers can be found in the engineering literature on the SE method [36–38, 48]. However, these papers all deal with DDEs where the time-varying coefficients are continuous functions of time. In contrast, milling is a system where the time-periodic parameters have state-independent discontinuities caused by the periodic entrance and exit of the cutting teeth. Due to these discontinuities, a further extension of the SE method is needed in order to apply it efficiently to milling models. In the following, the SE method is applied to (3.1) and it is shown that without the special treatment of jumps in the time-periodic coefficients, the numerical method is inefficient.

The governing equation (3.1) is a specific case of (2.39), thus its discretization can be carried out with the SE method according to Section 2.2. The discretized system (2.64) simplifies to

$$\begin{aligned} \mathbf{S}_{i,j}^0 \tilde{\mathbf{z}}_{k-1,n+1} + \sum_{j=2}^{n+1} \mathbf{S}_{i,j}^0 \tilde{\mathbf{z}}_{k,j} + \mathbf{R}_{i,j}^k \tilde{\mathbf{z}}_{k-1,n+1} - \mathbf{R}_{i,j}^k \tilde{\mathbf{z}}_{k-E-1,n+1} \\ + \sum_{j=2}^{n+1} \mathbf{R}_{i,j}^k (\tilde{\mathbf{z}}_{k,j} - \tilde{\mathbf{z}}_{k-E,j}) = \mathbf{0}, \quad i = 1, 2, \dots, n; \quad k = 1, 2, \dots, E; \end{aligned} \quad (3.11)$$

where the application of Lobatto-type Legendre-Gauss quadrature for integration and interpolation leads to sub-matrices

$$\mathbf{S}_{i,j}^0 = \int_{-1}^1 \left(\frac{2}{h} \mathbf{I} \phi_j'(\zeta) - \mathbf{A} \phi_j(\zeta) \right) \psi_i(\zeta) d\zeta = \frac{2}{h} \mathbf{I} \sum_{q=1}^{n+1} F_{i,q} D_{q,j} - \mathbf{A}_0 F_{i,j}, \quad (3.12)$$

$$\mathbf{R}_{i,j}^k = \int_{-1}^1 \mathbf{B} \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \phi_j(\zeta) \psi_i(\zeta) d\zeta \approx F_{i,j} \mathbf{B}_j^k. \quad (3.13)$$

Here $F_{i,q}$ and $D_{q,l}$ are defined according to Section 2.1.3 and 2.1.1, respectively and

$$\mathbf{B}_j^k = \mathbf{B} \left(\frac{h(\zeta_j^*+1)}{2} + (k-1)h \right). \quad (3.14)$$

Using (3.11), a mapping can be constructed according to (2.77)–(2.88), with $\mathbf{S}_{i,j}^k = \mathbf{S}_{i,j}^0 + \mathbf{R}_{i,j}^k$. Note that since $\Gamma = 1$, the identity part (2.77) of the matrix approximation $\mathbf{U} \in \mathbb{R}^{s(En+1) \times s(En+1)}$ of the monodromy operator shrinks to zero size. The computation of matrices Λ and Υ simplifies to

$$\Lambda = \Lambda^0 + \Lambda^{S_0} + \Lambda^R, \quad \Upsilon = \Upsilon^0 + \Upsilon^R, \quad (3.15)$$

where the structure of Λ^{S_0} , Λ^R and Υ^R is shown in Figure 3.1, while the structure of matrices Λ^0 and Υ^0 is given in (2.98).

Note that during the calculation of the SLDs, (3.14) has to be reevaluated in $E(n+1)$ number of quadrature points each time Ω is updated. The reason for this is that in (3.14) the argument of function $\mathbf{B}(t)$ incorporates $h = \tau/E = 60/(\Omega v E)$.

$$\Lambda^{S_0} = \begin{bmatrix} \mathbf{0} & & & \\ \mathbf{S}_{i,j}^0 & \ddots & & \\ & \ddots & \mathbf{S}_{i,j}^0 & \\ & & & \mathbf{S}_{i,j}^0 \end{bmatrix} \quad \Lambda^R = \Upsilon^R = \begin{bmatrix} \mathbf{0} & & & \\ \mathbf{R}_{i,j}^1 & \ddots & & \\ & \ddots & \mathbf{R}_{i,j}^2 & \\ & & & \mathbf{R}_{i,j}^E \end{bmatrix}$$

FIGURE 3.1: A schematic of the structure of matrices Λ^{S_0} , Λ^R and Υ^R .

m_t	0.03993 kg
ω_n	$922 \times 2\pi$ rad/s
ζ	0.011
q_c	1
K_t	6×10^8 N/m ²
K_n	2×10^8 N/m ²
v	2

TABLE 3.1: Parameters used throughout Section 3.1, chosen according to [32]

Since the reevaluation of (3.14) is time consuming, the dependence of (3.14) on Ω is need to be eliminated. This is done by the introduction of the angle domain with variable $\theta = 2\pi\Omega t/60$ instead of the time domain with variable t , which transforms (3.1) to

$$\xi'(\theta) = \hat{\mathbf{A}}_0 \xi(\theta) - \hat{\mathbf{B}}(\theta) \xi(\theta) + \hat{\mathbf{B}}(\theta) \xi(\theta - 2\pi/v), \quad (3.16)$$

where now \square' denotes differentiation with respect to θ and

$$\hat{\mathbf{B}}(\theta) = \sum_{p=1}^v \hat{\mathbf{B}}_p(\theta), \quad (3.17)$$

with matrices

$$\hat{\mathbf{A}}_0 = \begin{bmatrix} 0 & 1 \\ -1/\Omega_d^2 & -2\zeta/\Omega_d \end{bmatrix}, \quad \hat{\mathbf{B}}_p(\theta) = \frac{w_d}{\Omega_d^2} \begin{bmatrix} 0 & 0 \\ H_p(\theta) & 0 \end{bmatrix} \quad (3.18)$$

and

$$\hat{\mathbf{A}}_0 = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\frac{1}{\Omega_d^2} \mathbf{I} & -\frac{2\zeta}{\Omega_d} \mathbf{I} \end{bmatrix}, \quad \hat{\mathbf{B}}_p(\theta) = \frac{w_d}{\Omega_d^2} \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{H}_p(\theta) & \mathbf{0} \end{bmatrix} \quad (3.19)$$

for the single and two DoF models, respectively. The dimensionless spindle speed is denoted by $\Omega_d = 2\pi\Omega/(60\omega_n)$, while the dimensionless specific cutting force coefficient is $w_d = \tilde{w}/\omega_n^2$. After dropping the hats, the SE method is applied using the formulas above with dimensionless delay $\tau = 2\pi/v$.

Throughout Section 3.1, the results are presented for the parameter set given in Table 3.1. For down-milling, the results are shown in Figure 3.2. The left column of the figure shows that the low immersion ratio ($a_e/D = 0.05$) results in periodic jumps in $H(\theta)$ and that the convergence of the stability chart is very slow. Even with polynomial order $n = 50$, the stability boundary does not coincide with the exact one. In contrast, the right column of Fig. 3.2 shows that in the case of full

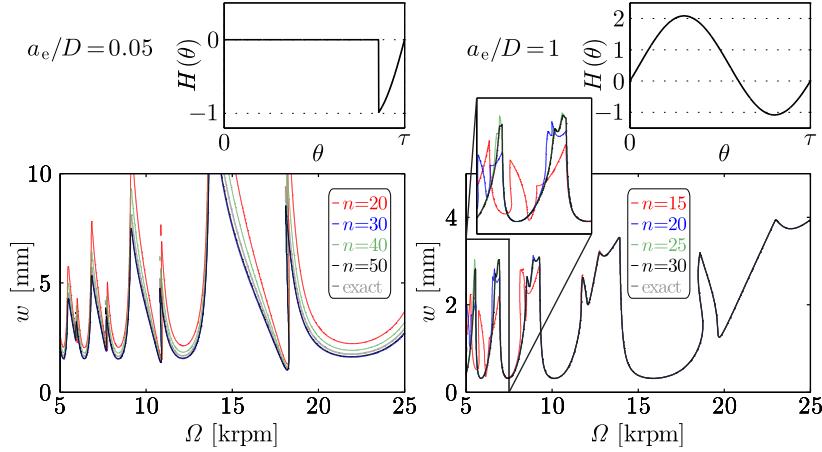


FIGURE 3.2: Convergence of stability maps for $s/2 = 1$ DoF, down-milling without any treatment of discontinuities in $\mathbf{B}(\theta)$. The radial immersion ratios are $a_e/D = 0.05$ and $a_e/D = 1$, the element number is $E = 1$ and the stability maps are calculated on a 400×400 grid. Further parameters are taken from Table 3.1.

immersion ($a_e/D = 1$) there is no discontinuity in $H(\theta)$ and that the convergence of the stability boundary is fast. In this case, polynomial order $n = 25$ already gives accurate results. It can be thus inferred that stability maps converge slowly when discontinuities are present in $H(\theta)$ (that is when $\mathbf{B}(\theta)$ is discontinuous), otherwise they converge fast. Consequently, in order to avoid poor convergence properties it is necessary to further improve the method to handle discontinuities. These improvements are detailed in the following.

3.1.3 Extension of the SE method

The inaccuracy in the stability calculation is caused by the inaccuracy of the Lobatto-type Legendre-Gauss quadrature in (3.13) due to the discontinuity of function $\mathbf{B}(\theta)$. In order to avoid discontinuity within $\mathbf{R}_{i,j}^k$, the domain of integration is split onto sub-domains where function $\mathbf{B}(\theta)$ remains smooth. Discontinuities are caused either by the entering or the exiting teeth of the tool. Consequently, in order to locate the discontinuities, it is useful to decompose $\mathbf{B}(\theta)$ into parts which correspond to different teeth. This leads to the expression

$$\mathbf{R}_{i,j}^k = \sum_{p=1}^v \mathbf{R}_{i,j}^{k,p} = \sum_{p=1}^v \int_{-1}^1 \mathbf{B}_p \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \phi_j(\zeta) \psi_i(\zeta) d\zeta. \quad (3.20)$$

The discontinuity points can be located for each $\mathbf{B}_p(\theta)$ function. Throughout this thesis, milling tools are assumed with uniformly distributed cutting teeth. In the following, parameters

$$\theta_1^p = \varphi_{\text{ent}} - (p-1) \frac{2\pi}{v}, \quad \theta_2^p = \varphi_{\text{ex}} - (p-1) \frac{2\pi}{v}, \quad (3.21)$$

are introduced for the notation of the angle of rotation corresponding to the entrance and exit of the p -th tooth, respectively. The possible cases for the location of the discontinuities relative to the principal period $[0, \tau]$ are the following:

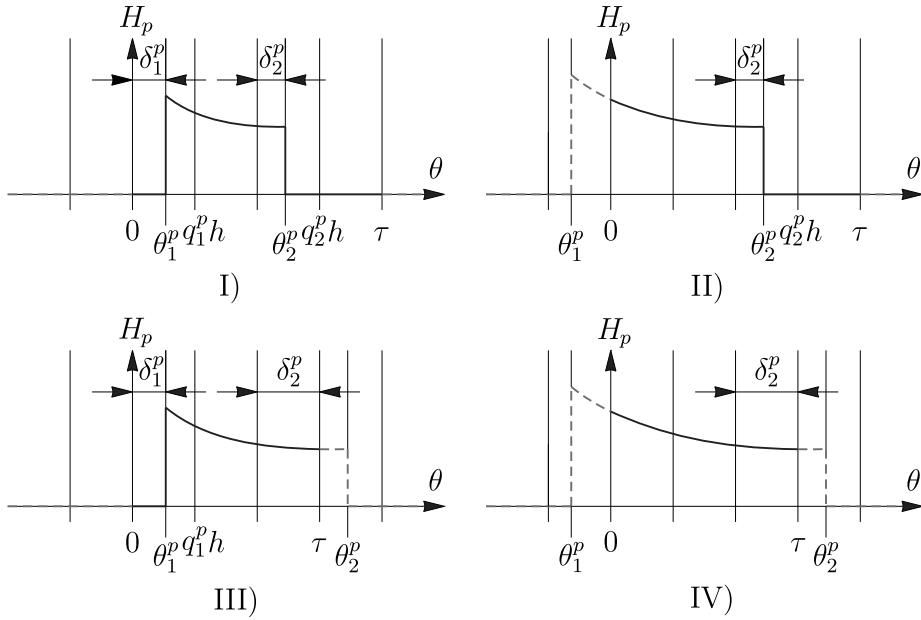


FIGURE 3.3: Different scenarios for the relation between the principal period and the entrance and exit of the p^{th} tooth, which define the discontinuities in $H_p(\theta)$.

I) $0 < \theta_1^p < \tau$ and $0 < \theta_2^p < \tau$

$$\begin{aligned}\delta_1^p &= \text{mod}(\theta_1^p, h), & q_1^p &= \text{floor}(\theta_1^p/h) + 1 \\ \delta_2^p &= \text{mod}(\theta_2^p, h), & q_2^p &= \text{floor}(\theta_2^p/h) + 1\end{aligned}$$

II) $\theta_1^p \leq 0$ and $0 < \theta_2^p < \tau$

$$\begin{aligned}\delta_1^p &= 0, & q_1^p &= 1 \\ \delta_2^p &= \text{mod}(\theta_2^p, h), & q_2^p &= \text{floor}(\theta_2^p/h) + 1\end{aligned}$$

III) $0 < \theta_1^p < \tau$ and $\theta_2^p \geq \tau$

$$\begin{aligned}\delta_1^p &= \text{mod}(\theta_1^p, h), & q_1^p &= \text{floor}(\theta_1^p/h) + 1 \\ \delta_2^p &= h, & q_2^p &= E\end{aligned}$$

IV) $\theta_1^p \leq 0$ and $\theta_2^p \geq \tau$

$$\begin{aligned}\delta_1^p &= 0, & q_1^p &= 1 \\ \delta_2^p &= h, & q_2^p &= E\end{aligned}$$

V) $\theta_1^p \leq 0$ and $\theta_2^p \leq 0$ (the p^{th} tooth is not in the cut)

$$\begin{aligned}\delta_1^p &= 0, & q_1^p &= 1 \\ \delta_2^p &= 0, & q_2^p &= 1\end{aligned}$$

VI) $\theta_1^r \geq \tau$ and $\theta_2^r \geq \tau$ (the p^{th} tooth is not in the cut)

$$\begin{aligned}\delta_1^p &= h, & q_1^p &= E \\ \delta_2^p &= h, & q_2^p &= E\end{aligned}$$

Only the first four cases are associated with actual cutting over the principal period $[0, \tau]$, these cases are illustrated in Figure 3.3. Using these formulas, the integral terms in (3.20) can be calculated as

$$\mathbf{R}_{i,j}^{k,p} = \begin{cases} \int_{-1+\beta_1^p}^{-1+\beta_2^p} \mathbf{B}_p \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \phi_j(\zeta) \psi_i(\zeta) d\zeta & \text{if } k = q_1^p = q_2^p, \\ \int_{-1}^{-1+\beta_2^p} \mathbf{B}_p \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \phi_j(\zeta) \psi_i(\zeta) d\zeta & \text{if } k = q_2^p \neq q_1^p, \\ \int_{-1+\beta_1^p}^1 \mathbf{B}_p \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \phi_j(\zeta) \psi_i(\zeta) d\zeta & \text{if } k = q_1^p \neq q_2^p, \\ \int_{-1}^1 \mathbf{B}_p \left(\frac{h(\zeta+1)}{2} + (k-1)h \right) \phi_j(\zeta) \psi_i(\zeta) d\zeta & \text{if } q_1^p < k < q_2^p, \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (3.22)$$

where $\beta_1^p = 2\delta_1^p/h$ and $\beta_2^p = 2\delta_2^p/h$. The integral terms in (3.22) are evaluated using the Lobatto-type Legendre-Gauss quadrature according to

$$\mathbf{R}_{i,j}^{k,p} = \begin{cases} \frac{\beta_2^p - \beta_1^p}{2} \sum_{q=1}^{n+1} F_{i,q}^I \mathbf{B}_{q,I}^{k,p} L_{q,j}^I & \text{if } k = q_1^p = q_2^p, \\ \frac{\beta_2^p}{2} \sum_{q=1}^{n+1} F_{i,q}^{II} \mathbf{B}_{q,II}^{k,p} L_{q,j}^{II} & \text{if } k = q_2^p \neq q_1^p, \\ \frac{2 - \beta_1^p}{2} \sum_{q=1}^{n+1} F_{i,q}^{III} \mathbf{B}_{q,III}^{k,p} L_{q,j}^{III} & \text{if } k = q_1^p \neq q_2^p, \\ F_{i,j} \mathbf{B}_{j,IV}^{k,p} & \text{if } q_1^p < k < q_2^p, \\ \mathbf{0} & \text{otherwise,} \end{cases} \quad (3.23)$$

where

$$\mathbf{B}_{j,\alpha}^{k,p} = \mathbf{B}_p \left(\frac{h(\eta_j^\alpha + 1)}{2} + (k-1)h \right), \quad L_{q,j}^\alpha = \phi_j(\eta_q^\alpha), \quad F_{i,q}^\alpha = \psi_i(\eta_q^\alpha) w_q, \quad (3.24)$$

and

$$\eta_j^\alpha = \begin{cases} \frac{\beta_2^r - \beta_1^r}{2} (\zeta_j + 1) - 1 + \beta_1^r & \text{if } \alpha = I, \\ \frac{\beta_2^r}{2} (\zeta_j + 1) - 1 & \text{if } \alpha = II, \\ \frac{2 - \beta_1^r}{2} (\zeta_j + 1) - 1 + \beta_1^r & \text{if } \alpha = III, \\ \zeta_j & \text{if } \alpha = IV. \end{cases} \quad (3.25)$$

As it is evident from Fig. 3.4, formula (3.20) with (3.23) gives fast convergence for the stability boundaries even in the presence of discontinuities in $\mathbf{B}(\theta)$. For example, for spindle speed range $\Omega \in [5000, 25000]$ rpm, polynomial order $n = 20$ already gives accurate results for radial immersion ratios $a_e/D = 0.05, 0.1$ and 0.5 . For immersion ratios $a_e/D = 0.75$ and 1 , accurate calculations require slightly higher polynomial orders ($n = 30$ and $n = 25$, respectively). Note that, for $v = 2$, the smaller the radial immersion, the longer the free vibration. The solution in the free vibration period is more accurate for a fixed polynomial order n than the approximation of the solution segment during cutting. Consequently, the higher the ratio of the duration of the free vibration and cutting, the more accurate the solution. Also, note that there exists another solution for the treatment of discontinuities by locating the element boundaries at the discontinuity points of $\mathbf{B}(\theta)$ (see

[8, 56]). The herein presented approach provides a generalized framework, which handles discontinuities without adjusting the number and the length of the elements.

3.1.4 Comparison with other methods

Since in practice it is desirable to apply the computationally most efficient approach, it is necessary to compare the SE method with other well-known numerical methods from the literature. In order to enable the comparison of computational efficiency, the same examples are chosen from the published studies [18–21, 32, 55, 66, 70] for the single and two DoF models of milling processes. The computational efficiency of the SE method is investigated and compared based on three criteria: the computational time, the convergence of stability boundaries, and the convergence rate of the largest characteristic multiplier. The Matlab code used for the calculation of stability boundaries can be downloaded from <http://www.mm.bme.hu/~lehotzky/IJAMT2016>.

Single DoF model. Figures 3.2, 3.4 and 3.5 show stability diagrams which correspond to radial immersion ratios $a_e/D = 0.05, 0.1, 0.5, 0.75$ and 1. Clearly, in the case of continuous $B(\theta)$ (such as for full-immersion milling), formulas with and without the treatment of discontinuities give the same results. The figures show that polynomial orders $n = 20 \sim 30$ give accurate results for the stability boundaries on the domain $\Omega \in [5000, 25000]$ rpm. It is also interesting to note that for lower spindle speeds higher polynomial order n is required to achieve the same accuracy. This can be seen in Figure 3.5, where the stability map of full immersion down-milling is depicted for lower spindle speed ranges. It can also be inferred that longer non-zero continuous part of $B(\theta)$ (that is longer time within the cut) requires slightly higher polynomial order n for accurate results.

In order to show the efficiency of the extended SE method, the computational time of constructing stability diagrams is compared to those in the literature. For this purpose, cases which cover the results presented in [18–20, 32, 55, 66, 70] are selected. In order to perform meaningful comparisons, the stability diagrams were obtained using a computer with similar specifications to the computers used in the above references. Namely, a PC running Matlab 2009 with 2.1 GHz Core 2 Duo processor and 2GB RAM memory. The consistency of the computational hardware allows performing direct comparisons between the SE and other prominent methods in the literature as shown in Table 3.2. Specifically, Table 3.2 summarizes the computational times corresponding to these comparisons. Reference results using the zeroth order, updated semi-discretization (SD) method (see [32] for details) were calculated and their computational times corresponding to accurate stability boundaries are given in the table. These results were obtained using sparse matrix multiplication in the Matlab code for better computational efficiency on the above specified PC. Note that contrary to the SE method, for the SD method, n denotes the number of divisions used in the history segment. In addition to the reference results obtained by the SD method, the lowest computational times are also collected from [18–20, 32, 55, 66, 70] for comparison with the SE method. In these references, different spindle speed domains and immersion ratios are investigated on different grids of the parameter plane. These are all specified in Table 3.2 in order to facilitate precise and fair comparison. The domains for the depth of cut w are selected according to the corresponding stability maps in Figures 3.2, 3.4–3.5 and 3.8. These figures were also used for the selection of polynomial order n in Table 3.2, where

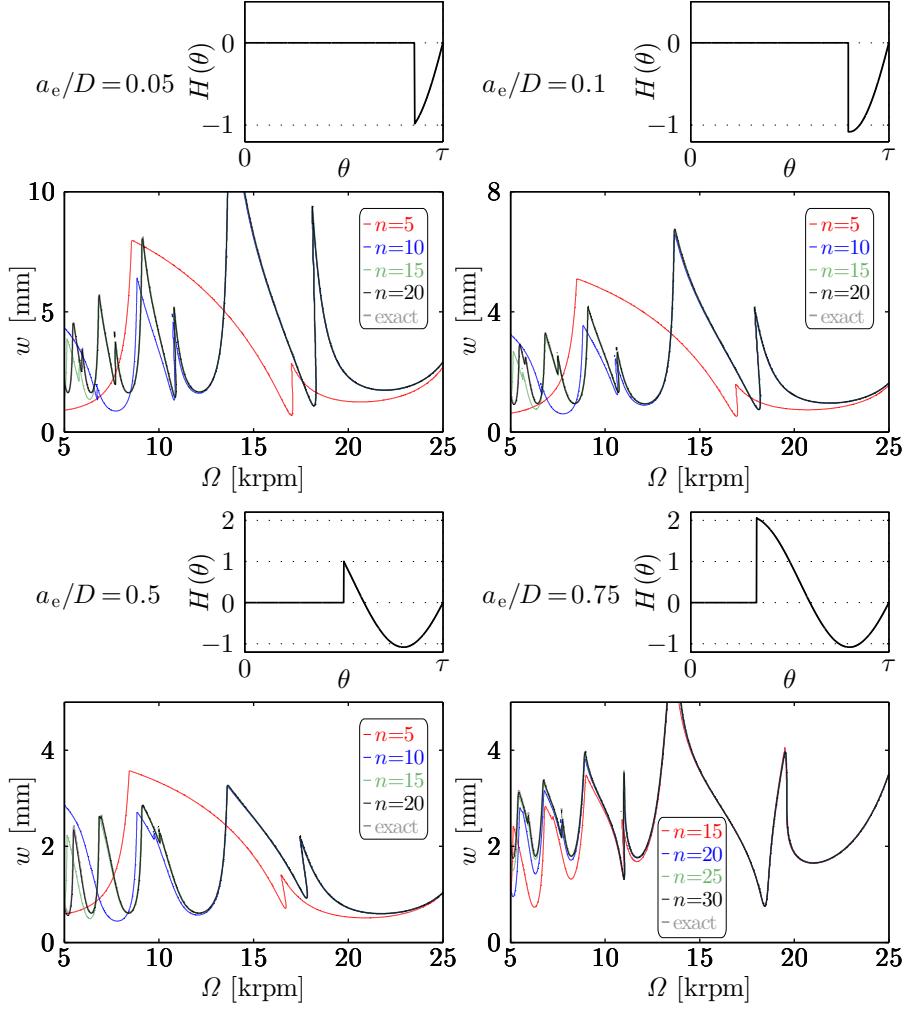


FIGURE 3.4: Convergence of stability maps for $s/2 = 1$ DoF down-milling operation with the treatment of discontinuities in $\mathbf{B}(\theta)$ for radial immersion ratios $a_e/D = 0.05, 0.1, 0.5, 1$. The element number is $E = 1$ and the stability maps are calculated on a 400×400 grid. Further parameters are taken from Table 3.1.

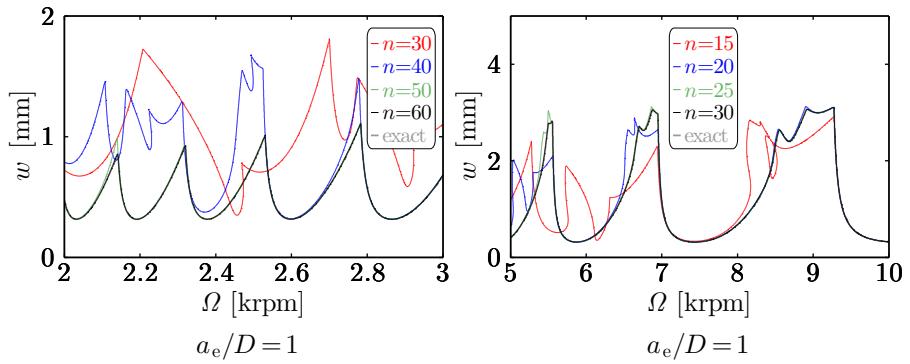


FIGURE 3.5: Convergence of stability maps for $s/2 = 1$ DoF down-milling operation with the treatment of discontinuities in $\mathbf{B}(\theta)$ for low spindle speed ranges. The radial immersion ratio is $a_e/D = 1$, the element number is $E = 1$ and the stability maps are calculated on a 400×400 grid. Further parameters are taken from Table 3.1.

Grid	$\Omega \in$	a_e/D	1 DoF model					2 DoF model				
			n		Computational time [s]			n		Computational time [s]		
			SE	SD	SE	SD	Reference	SE	SD	SE	SD	Reference
400 × 200	[5, 25] krpm	0.05	20	40	32.35	776.4	13.9 [19]	20	40	174.3	1105	960.6 [18]
		0.1	20	40	31.96	798.5	293.1 [18]	20	30	177.9	840.1	954.1 [18]
		0.5	20	40	32.32	824.5		20	20	136.5	441.4	
		1	25	50	54.49	1237	129.6 [19]	20	20	138.5	505.9	
200 × 100	[5, 10] krpm	0.05	20	40	8.78	190.7		20	40	42.44	279.5	
		0.1	20	40	8.57	187.0		20	30	41.45	190.7	
		0.5	20	40	8.56	183.9		20	20	33.63	108.2	
		1	25	50	14.71	274.1	49 [55]	20	20	32.76	122.3	
100 × 50	[5, 25] krpm	0.05	20	40	2.41	44.80		20	40	11.31	67.52	26.5 [66]
		0.1	20	40	2.44	44.49		20	30	11.16	45.36	
		0.5	20	40	2.46	46.23		20	20	8.94	27.84	9.3 [66]
		1	25	50	4.04	69.79		20	20	8.69	32.03	27.4 [66]

TABLE 3.2: Computational times for different milling models for different radial immersion ratios on three different grids and different spindle speed ranges, with a total number of elements $E = 1$. Further parameters are taken from Table 3.1. For the single DoF model down-milling for the two DoF model up-milling was considered.

polynomial orders corresponding to accurate results on the specified spindle speed ranges are given.

Table 3.2 shows that the SE method is the least computationally extensive approach in comparison with most of the other methods. One reason for the low computational time of the SE method is that only one matrix inversion and one matrix multiplication is needed to obtain the monodromy matrix. This contrasts the need for multiple matrix inversions which is necessary, for example, when using the semi- and full-discretization methods.

Another reason for the efficiency of the SE method is its exponential convergence rate with respect to polynomial order n . The exponential convergence rate is shown in Figure 3.6, where the absolute error of the critical characteristic multiplier $\tilde{\mu}_{\text{cr}}$, calculated by the SE method, is shown as a function of polynomial order n for fixed number of elements E . The absolute error of $\tilde{\mu}_{\text{cr}}$ is calculated with respect to the reference characteristic multiplier μ_{cr} which was determined using the SE method with $E = 10$ and $n = 50$. It can be seen in the figure that the absolute error is progressively decreasing with the increase of polynomial order n on a logarithmic scale. Comparing the convergence rates in [19, 20, 55, 66, 70] with the corresponding rates for the SE method in Figure 3.6 shows that the latter converges faster with respect to n . The convergence rates of the SE method are similar to those in [21]. However, since the PC used for obtaining the results in [21] has better computational power, the corresponding efficiency of calculations cannot be meaningfully compared to the rest of the methods in Table 3.2. The only case in Table 3.2 where the computational time is smaller than that of the SE method is the low immersion single degree of freedom milling with $a_e/D = 0.05$. Here, the numerical

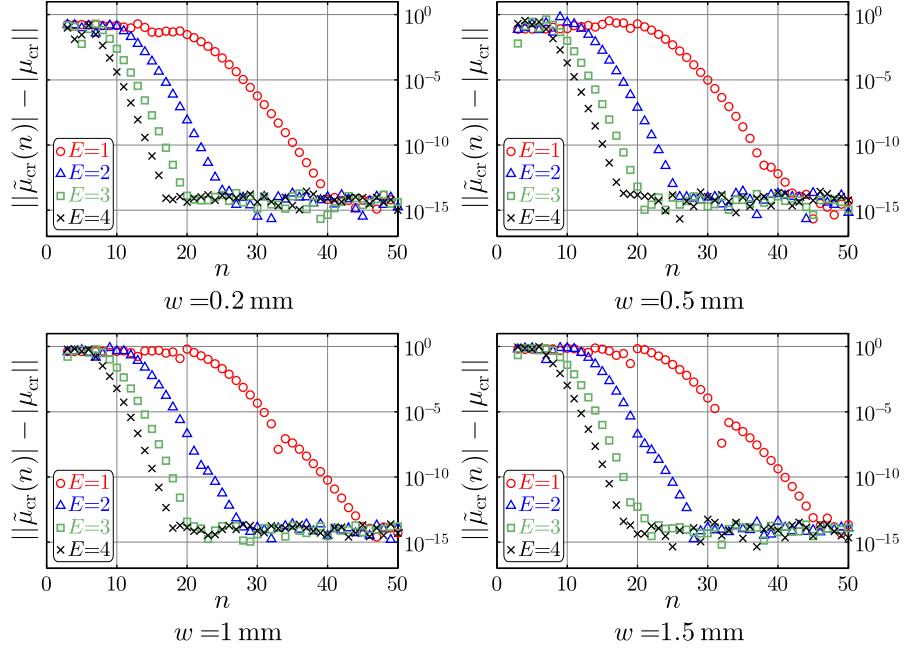


FIGURE 3.6: Absolute error of the critical characteristic multiplier with respect to the reference critical multiplier μ_{cr} calculated for the case $E = 10$ and $n = 50$. The absolute error is shown as a function of polynomial order n for different element numbers E . Results are depicted for $s/2 = 1$ DoF down-milling operations with radial immersion ratio $a_e/D = 1$, spindle speed $\Omega = 5000$ rpm and depths of cut $w = 0.2$, $w = 0.5$ mm, $w = 1$ mm and $w = 1.5$ mm. Further parameters are taken from Table 3.1.

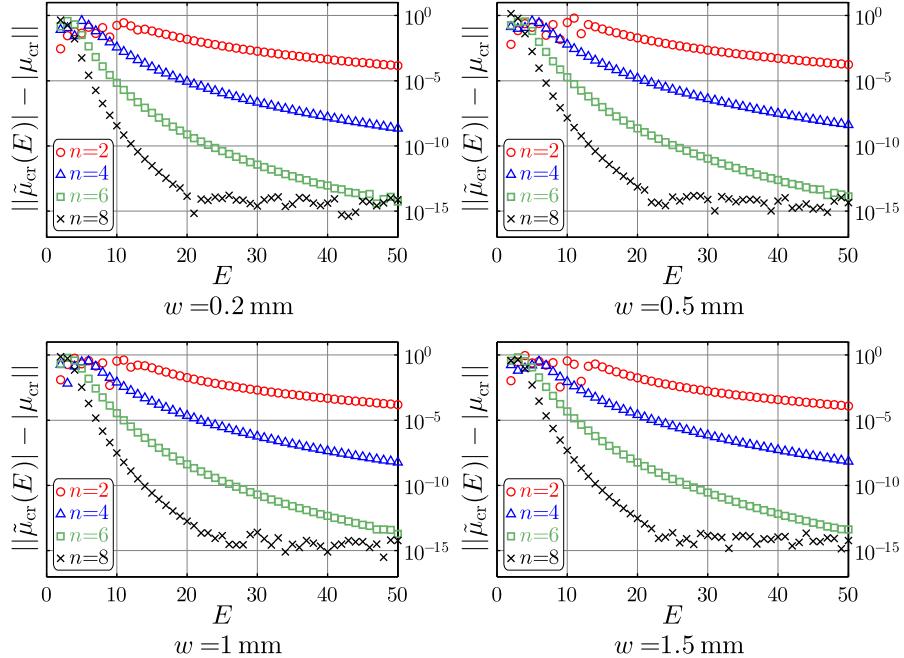


FIGURE 3.7: Absolute error of the critical characteristic multiplier with respect to the reference critical multiplier μ_{cr} calculated for the case $E = 10$ and $n = 50$. The absolute error is shown as a function of element number E for different polynomial orders n . Results are depicted for $s/2 = 1$ DoF down-milling operation with radial immersion ratio $a_e/D = 1$, spindle speed $\Omega = 5000$ rpm and depths of cut $w = 0.2$, $w = 0.5$ mm, $w = 1$ mm and $w = 1.5$ mm. Further parameters are taken from Table 3.1.

integration method [19] utilizes the closed form solution during the free oscillation. In contrast, our method does not distinguish between free oscillation and cutting, and solves the free oscillation using the SE method. Note that the separation of free vibration could also be applied to the SE method in order to decrease the computation time, however, here the goal was to give a general framework rather than utilizing particular properties of the system. To elaborate, free vibration is present only for a tool with low number of teeth in case of low-radial-immersion milling. Nevertheless, the computation time for full-immersion milling (where there is no free oscillation) is less using the SE method than the numerical integration method.

Note that the SE method has two approximation parameters. One is the above discussed polynomial order n , the other one is the number of elements E . In Figure 3.7, the convergence rate of the absolute error of the critical characteristic multiplier is depicted on a logarithmic scale as a function of the number of elements E for different polynomial orders n . It can be seen that the method does not have exponential convergence with respect to E .

Two DoF model. The convergence of the stability maps are shown in Figure 3.8 for up-milling operations with different radial immersion ratios. The calculations are carried out with different polynomial orders n . It can be seen that polynomial order $n = 20$ already provides accurate results over spindle speed domain $\Omega \in [5000, 25000]$ rpm for all radial immersion ratios. The computational times for different computational arrangements can be found in Table 3.2, where the computational times of the SD method and the lowest computational times from references [18–20, 32, 55, 66, 70] are compared to the results of the SE method. The results show that the SE method provides the fastest computational times among all of the investigated cases.

3.2 Digital position control in machining

As it was anticipated at the beginning of this chapter, for a better utilization of the capacity of the machining center active chatter suppression techniques can be applied. One of these techniques, called active damping [65], builds an additional control loop to the machine tool close to the tool tip, where it actuates based on the measured velocity and displacement. Note that in machining centers the feed motion of the tool relative to the workpiece is provided by the controllers of feed drives. Therefore controllers are naturally present in machining centers and affect the machining process through the feed motion. In this chapter, mechanical models are derived, which incorporate the control loop of the active damper or of the feed drives of the machining center in the model of the regenerative cutting process. For the modeling of feed drives two different cases are investigated. First, it is assumed that the tool holder is fixed and the workpiece holder is moved by a feed drive. Then the workpiece holder is considered to be fixed and the tool holder is moved by a feed drive. For all cases, digital position control is applied according to the control scheme, presented in Section 2.4. After the derivation of the mechanical models, their linear stability is analyzed about periodic motions using the numerical algorithms, proposed in Section 2.4.

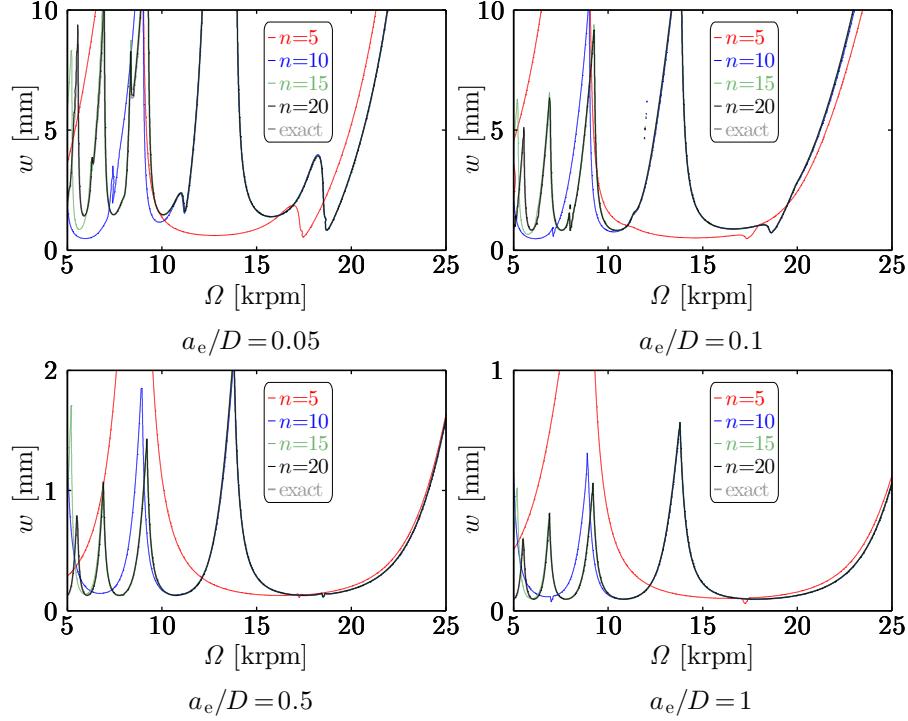


FIGURE 3.8: Convergence of stability maps for $s/2 = 2$ DoF up-milling operation with the treatment of discontinuities in $\mathbf{B}(\theta)$ for radial immersion ratios $a_e/D = 0.05, 0.1, 0.5, 1$. The element number is $E = 1$ and the stability maps are calculated on a 400×400 grid. Further parameters are taken from Table 3.2.

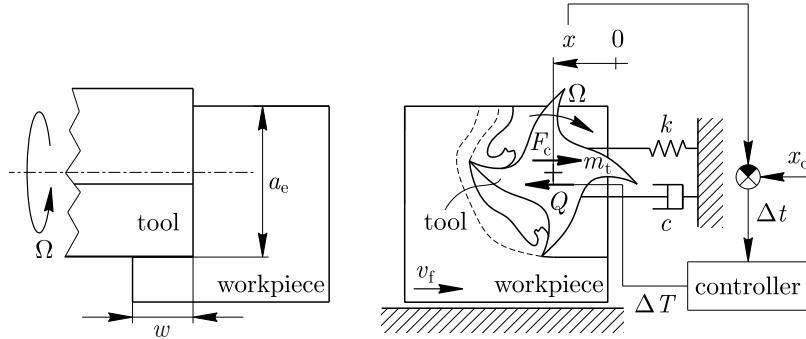


FIGURE 3.9: Model of milling operations subjected to active damping

3.2.1 Milling process with active damper

The one DoF mathematical model, presented in Section 3.1.1 assumes that the tool (or the workpiece) can oscillate in the direction of the feed velocity only. In addition, the mechanical model, shown in Figure 3.9, takes into account a feedback loop controlled by a proportional-integral-derivative (PID) controller, which provides the active damping to the milling process. The control force Q thus consists of integral, proportional and derivative terms with feedback gains I , P and D , respectively. The tool is modeled by a block of mass m_t which is connected to the tool holder via a spring of stiffness k and a dashpot of viscous damping c as shown in Figure 3.9. The workpiece is assumed to move horizontally with a constant feed velocity v_f relative to the tool holder. The undamped natural angular frequency of

the tool is $\omega_n = \sqrt{k/m_t}$ and the damping ratio is $\zeta = c/(2m_t\omega_n)$. Using dimensionless time $\hat{t} = \omega_n t$ and dropping the hat immediately, the governing equations are

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}_0 \mathbf{x}(t) + \mathbf{C}(\mathbf{x}(t_l - \delta t) - \mathbf{x}_d(t_l - \delta t)) \\ &\quad + \mathbf{E} \mathbf{X}_l - \mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d)) \quad t \in [t_l, t_{l+1}],\end{aligned}\quad (3.26)$$

$$\mathbf{X}_l = \mathbf{X}_{l-1} + \sum_{b=1}^{\tilde{n}} \mathbf{W}_b (\mathbf{x}(t_l - b\delta t) - \mathbf{x}_d(t_l - b\delta t)), \quad (3.27)$$

where the state $\mathbf{x}(t)$ and the desired trajectory $\mathbf{x}_d(t)$ are defined as

$$\mathbf{x}(t) = \begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix}, \quad \mathbf{x}_d(t) = \begin{bmatrix} x_d(t) \\ \dot{x}_d(t) \end{bmatrix}, \quad (3.28)$$

the numerical integral of $\mathbf{x}(t)$ at dimensionless time instant t_l is denoted by \mathbf{X}_l and

$$\mathbf{A}_0 = \begin{bmatrix} 0 & 1 \\ -1 & -2\zeta \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 0 & 0 \\ -k_P & -k_D \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} 0 & 0 \\ -k_I & 0 \end{bmatrix}, \quad (3.29)$$

$$\mathbf{W}_b = \begin{bmatrix} W_b & 0 \\ 0 & W_b \end{bmatrix}, \quad \mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d)) = \frac{F_c(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d))}{m_t \omega_n^2} \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \quad (3.30)$$

Due to the rescaled time, dimensionless sampling period $\delta t = \omega_n \Delta t$ and dimensionless control gains are introduced as $k_I = I/(m_t \omega_n^3)$, $k_P = P/(m_t \omega_n^2)$ and $k_D = D/(m_t \omega_n)$. The integral \mathbf{X}_l of the state is computed using quadrature weights W_b . In the following examples these quadrature weights are calculated according to the closed type Newton-Cotes formulas (see Section 25.4 in [1]). The dimensionless regenerative delay (which coincides with the tooth passing period) is $\tau_d = 2\pi/(v\Omega_d)$, with $\Omega_d = 2\pi\Omega/(60\omega_n)$ being the dimensionless spindle speed and v being the number of cutting teeth. The cutting force is the resultant of forces acting on the teeth (see Figure 3.10/B), hence the horizontal component of the resultant cutting force is

$$\begin{aligned}F_c(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d)) &= \\ &\sum_{p=1}^v g_p(t) (F_{t,p}(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d)) \cos(\varphi_p(t)) + F_{n,p}(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d)) \sin(\varphi_p(t))),\end{aligned}\quad (3.31)$$

where the window function is defined according to (3.6). The tangential and normal force components of the p^{th} tooth (shown in Figure 3.10/B) are both calculated according to the power law (see [39] for details)

$$F_{t,p}(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d)) = w K_t h_p^{q_c}(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d)), \quad (3.32)$$

$$F_{n,p}(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d)) = w K_n h_p^{q_c}(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d)), \quad (3.33)$$

respectively. Here the angular position of the p^{th} tooth at dimensionless time instant t is $\varphi_p(t) = \Omega_d t + (p-1)2\pi/v$. By assuming circular tooth path (see Figure 3.10/A), the chip thickness on the p^{th} tooth at dimensionless time instant t can be

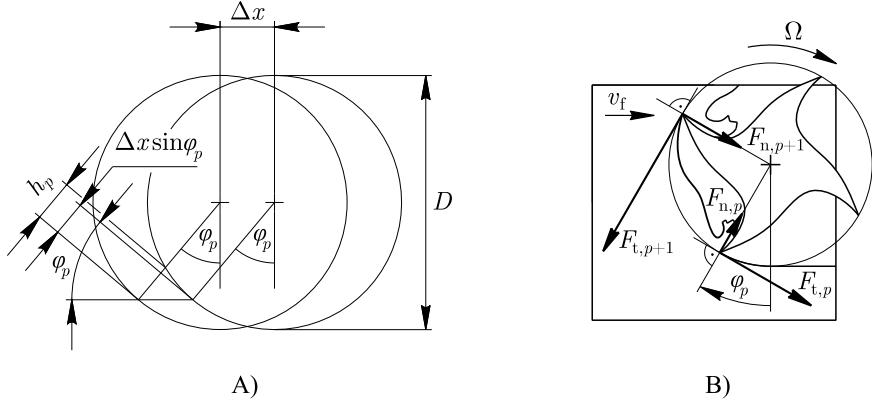


FIGURE 3.10: Cutting model: A) circular tooth path approximation B) tangential and normal cutting force components

approximated as

$$h_p(t, \mathbf{x}(t), \mathbf{x}(t-\tau_d)) \approx \Delta x(\mathbf{x}(t), \mathbf{x}(t-\tau_d)) \sin(\varphi_p(t)), \quad (3.34)$$

where

$$\Delta x(\mathbf{x}(t), \mathbf{x}(t-\tau_d)) = f_N + x(t) - x(t-\tau_d) \quad (3.35)$$

is the difference in the relative position of the tool and the workpiece between two consecutive cuts.

It is assumed that the stationary solution $(\bar{\mathbf{x}}(t), \bar{\mathbf{X}}_l)$ for (3.26)–(3.27) is equal to the desired solution $(\mathbf{x}_d(t), \mathbf{0})$, which gives

$$\dot{\bar{\mathbf{x}}}(t) = \mathbf{A}_0 \bar{\mathbf{x}}(t) - \mathbf{f}(t, \bar{\mathbf{x}}(t), \bar{\mathbf{x}}(t-\tau_d)). \quad (3.36)$$

Since $\mathbf{f}(t + \tau_d, \cdot, \cdot) = \mathbf{f}(t, \cdot, \cdot)$, there exists a τ_d -periodic stationary solution $\mathbf{x}_p(t) = \mathbf{x}_p(t + \tau_d)$ of (3.36). If the desired solution is set to be the periodic stationary solution of (3.36), that is $\mathbf{x}_d(t) = \mathbf{x}_p(t)$, then $\mathbf{f}(t, \mathbf{x}_p(t), \mathbf{x}_p(t-\tau_d)) = \mathbf{f}_p(t)$ becomes solely time-dependent (see (3.30)–(3.35)). Consequently, (3.36) becomes a linear autonomous ordinary differential equation for which the τ_d -periodic stationary solution $\mathbf{x}_p(t)$ can be computed in a simple way. Here, it is important to note that in practice $\mathbf{x}_p(t)$ and, consequently, $\bar{\mathbf{x}}(t)$ cannot be determined accurately due to modeling and parameter uncertainties, therefore the desired solution of the controller cannot be set equal to the stationary solution $\bar{\mathbf{x}}(t)$. This implies that the actual stationary solution is different from $\bar{\mathbf{x}}(t)$ and is determined by the nonlinear hybrid DDE-DE system (3.26)–(3.27). Nevertheless, here it is assumed that $\mathbf{x}_d(t) = \mathbf{x}_p(t)$ is known exactly as the τ_d -periodic stationary solution of (3.36). Consequently, linear stability properties can be analyzed by the variational system of (3.26)–(3.27) about the stationary solution $(\mathbf{x}_p(t), \mathbf{0})$.

In (3.26)–(3.27), the decomposition of state variables as $(\mathbf{x}(t), \mathbf{X}_l) = (\mathbf{x}_p(t) + \xi(t), \mathbf{0} + \chi_l)$ and the first-order Taylor expansion of $\mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t-\tau_d))$ about $\mathbf{x}_p(t)$ with respect to perturbation $\xi(t)$ leads to the variational system

$$\dot{\xi}(t) = (\mathbf{A}_0 - \mathbf{B}(t))\xi(t) + \mathbf{B}(t)\xi(t-\tau_d) + \mathbf{C}\xi(t_l - \delta t) + \mathbf{E}\chi_l \quad t \in [t_l, t_{l+1}), \quad (3.37)$$

$$\chi_l = \chi_{l-1} + \sum_{b=1}^{\tilde{n}} \mathbf{W}_b \xi(t_l - b\delta t), \quad (3.38)$$

where

$$\mathbf{B}(t) = \begin{bmatrix} 0 & 0 \\ w_d & 0 \end{bmatrix} \sum_{p=1}^v H_p(t), \quad (3.39)$$

and $H_p(t)$ is given according to (3.4). The stability of (3.37)–(3.38) can be analyzed according to Section 2.4. Note that $\mathbf{x}_p(t)$ is not involved in the variational system (3.37)–(3.38), therefore the periodic solution of (3.36) does not need to be determined. It is also important to note that similarly as it was presented in Section 3.1, $\mathbf{B}(t)$ is discontinuous in general. Consequently, during the application of the SE method, the extension presented in Section 3.1.3 is used.

The period of functions $g_p(t)$ is τ_d and the digital control introduces an additional time period: the dimensionless actuation period $\delta T = \tilde{n}\delta t$. Consequently, (3.37)–(3.38) is a quasi-periodic system. Here, it is assumed that the ratio of the actuation period and the tooth-passing period is rational and a principal period can be given as $T_p = \epsilon\delta T = \rho\tau_d$, with $\epsilon, \rho \in \mathbb{Z}^+$. Thus, the case given in Section 2.4 can be applied.

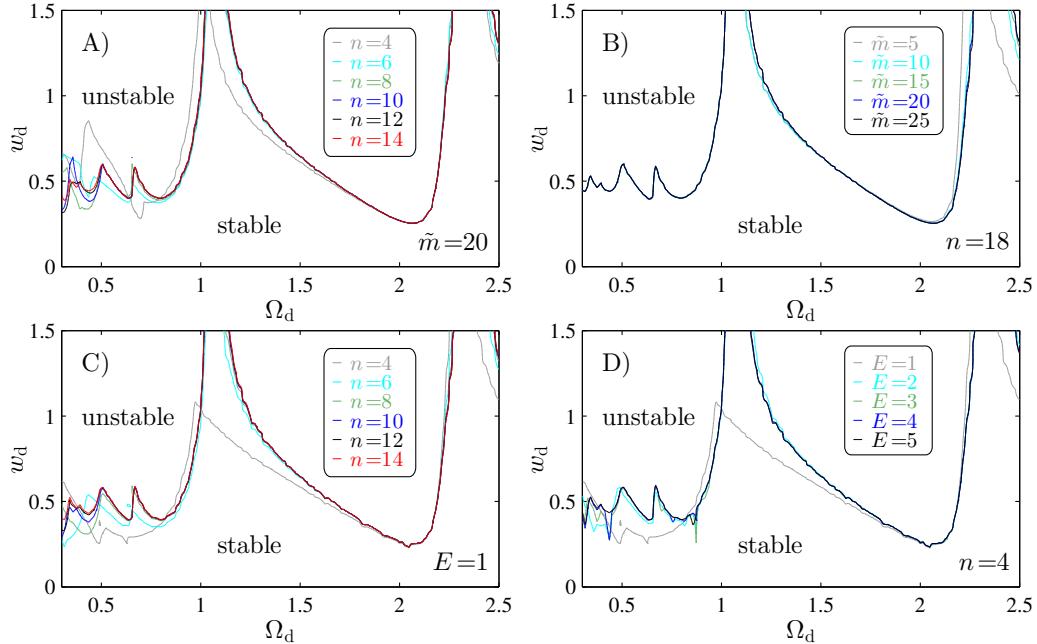


FIGURE 3.11: Stability lobe diagrams for down-milling operations with active damper using parameters $\zeta = 0.05$, $K_r = 3$, $a_e/D = 0.5$, $v = 2$, $q_c = 0.75$, $k_I = 0.1$, $k_P = 0.2$, $k_D = 0.2$, $\delta t = 0.1$ and $\tilde{n} = 4$. Stability diagrams A) and B) were calculated using the PsT method with fixed resolution \tilde{m} and order n of polynomial approximation, respectively. Stability diagrams C) and D) were calculated using the SE method with fixed element number E and order n of polynomial approximation, respectively.

The SLD approximations corresponding to (3.37)–(3.38) are shown in Figure 3.11 for a fixed parameter set given in the caption. In panels A) and B), the stability diagrams were computed using the PsT method. There are two approximation parameters in the PsT method: order n of polynomial approximation and resolution \tilde{m} of the dimensionless actuation period δT . The stability boundary converges if both n and \tilde{m} are increased, therefore sufficiently high n and \tilde{m} are required for an accurate stability boundary. In panels C) and D), the stability diagrams were computed using the SE method. There are two approximation parameters in the

SE method: order n of polynomial approximation and element number E , which is the number of elements within one dimensionless tooth passing period τ_d . The stability boundary converges if either n or E are increased, therefore sufficiently high n or E are required for an accurate stability boundary.

Note that the difficulty of the SLD calculation is given by the change in the tooth passing period $\tau_d = 2\pi/(\Omega_d v)$ along the axis Ω_d . Since dimensionless actuation period δT is fixed, this results in different $T_p = \epsilon\delta T = \rho\tau_d$ principal periods for each point along the axis Ω_d . Therefore the distribution of gridpoints along Ω_d is not uniform, because $\epsilon, \rho \in \mathbb{Z}^+$ has to hold.

3.2.2 Milling with controlled workpiece holder

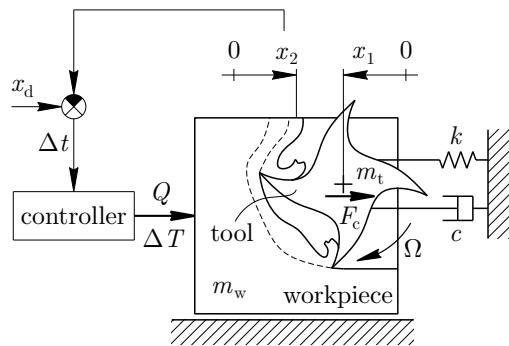


FIGURE 3.12: Model of milling operations with digitally controlled workpiece holder

In machining centers, the relative motion between the tool holder and the workpiece is provided by controlled feed drives, which usually use a feedback loop in order to stabilize the relative motion about the desired nominal feed velocity v_f . In this example, it is assumed that the tool holder is fixed while the position of the workpiece is controlled by a digital PID controller. The corresponding mechanical model is shown in Figure 3.12. In contrast with Figure 3.9, here the velocity of the workpiece is not constant but its motion is given by state variable $x_2(t)$. The control force Q is calculated based on the the error between the actual $x_2(t)$ and the desired $x_d(t) = v_f t$ position of the workpiece. The same way as in Section 3.2.1, the time is scaled as $\hat{t} = \omega_n t$, with $\omega_n = \sqrt{k/m_t}$ being the undamped natural angular frequency of the tool. After dropping the hat, the governing equations read

$$\dot{\mathbf{x}}(t) = \mathbf{A}_0 \mathbf{x}(t) + \mathbf{C} \mathbf{x}(t_l - \delta t) + \mathbf{E} \mathbf{X}_l - \mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t - \tau_d)) \quad t \in [t_l, t_{l+1}), \quad (3.40)$$

$$\mathbf{X}_l = \mathbf{X}_{l-1} + \sum_{b=1}^{\tilde{n}} \mathbf{W}_b \mathbf{x}(t_l - b\delta t), \quad (3.41)$$

where now

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) \\ \dot{x}_1(t) \\ x_2(t) - \hat{v}_f t \\ \dot{x}_2(t) - \hat{v}_f \end{bmatrix}, \quad \mathbf{A}_0 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & -2\zeta & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{C} = -\frac{1}{m_r} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & k_P & k_D \end{bmatrix}, \quad (3.42)$$

$$\mathbf{E} = -\frac{1}{m_r} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & k_I & 0 \end{bmatrix}, \quad \mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t-\tau_d)) = \frac{F_c(t, \mathbf{x}(t), \mathbf{x}(t-\tau_d))}{m_t \omega_n^2} \begin{bmatrix} 0 \\ 1 \\ 0 \\ \frac{1}{m_r} \end{bmatrix} \quad (3.43)$$

and $\mathbf{W}_b = W_b \mathbf{I}$, with $m_r = m_w/m_t$ being the ratio of the mass of the workpiece and the tool. The horizontal component F_c of the resultant cutting force is defined in the same way as in Section 3.2.1 except that the difference in the relative position of the tool and the workpiece between two consecutive cuts is changed from (3.35) to

$$\Delta x(\mathbf{x}(t), \mathbf{x}(t-\tau_d)) = x_1(t) + x_2(t) - (x_1(t-\tau_d) + x_2(t-\tau_d)). \quad (3.44)$$

The rest of the parameters in (3.40)–(3.43) are defined as in Section 3.2.1. Similarly to Section 3.2.1, it is assumed that there exists a periodic stationary solution $(\mathbf{x}_p(t), \mathbf{X}_l^p)$ for (3.40)–(3.41). Here it is important to note that in contrast with (3.36), the T_p -periodic solution $\mathbf{x}_p(t)$ cannot be simplified to a τ_d -periodic motion thus $\mathbf{f}(t, \mathbf{x}_p(t), \mathbf{x}_p(t-\tau_d))$ remains nonlinear with respect to $\mathbf{x}_p(t)$. With the assumption that periodic stationary solution $(\mathbf{x}_p(t), \mathbf{X}_l^p)$ exists the state variables can be decomposed as $(\mathbf{x}(t), \mathbf{X}_l) = (\mathbf{x}_p(t) + \boldsymbol{\xi}(t), \mathbf{X}_l^p + \boldsymbol{\chi}_l)$ and the first-order Taylor expansion of $\mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t-\tau_d))$ about $\mathbf{x}_p(t)$ with respect to perturbation $(\boldsymbol{\xi}(t), \boldsymbol{\chi}_l)$ gives a variational system of the form (3.37)–(3.38), where now matrices \mathbf{A}_0 , \mathbf{C} , \mathbf{E} and \mathbf{W}_b are defined according to (3.42)–(3.43), while $\boldsymbol{\xi}(t)$ and $\mathbf{B}(t)$ are given as

$$\boldsymbol{\xi}(t) = \begin{bmatrix} x_1(t) - x_1^p(t) \\ \dot{x}_1(t) - \dot{x}_1^p(t) \\ x_2(t) - x_2^p(t) \\ \dot{x}_2(t) - \dot{x}_2^p(t) \end{bmatrix}, \quad (3.45)$$

$$\mathbf{B}(t) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{m_r} & 0 & \frac{1}{m_r} & 0 \end{bmatrix} w_d f_N^{1-q_c} (x_1^p(t) - x_1^p(t-\tau_d) + x_2^p(t) - x_2^p(t-\tau_d))^{q_c-1} \sum_{p=1}^v H_p(t). \quad (3.46)$$

Here, in contrast with the previous example, the periodic stationary solution $(\mathbf{x}_p(t), \mathbf{X}_l^p)$ of (3.40)–(3.41) needs to be determined since now the components $x_1^p(t)$, $x_2^p(t)$ of $\mathbf{x}_p(t)$ are present in the variational system (3.37)–(3.38) through (3.46). For the sake of simplicity, it is assumed that $\tau_d = \tilde{\epsilon} \delta T$ with $\tilde{\epsilon} \in \mathbb{Z}^+$, which results in a

τ_d -periodic stationary solution $(\mathbf{x}_p(t), \mathbf{X}_l^p)$, and simplifies (3.46) to

$$\mathbf{B}(t) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{m_r} & 0 & \frac{1}{m_r} & 0 \end{bmatrix} w_d \sum_{p=1}^v H_p(t). \quad (3.47)$$

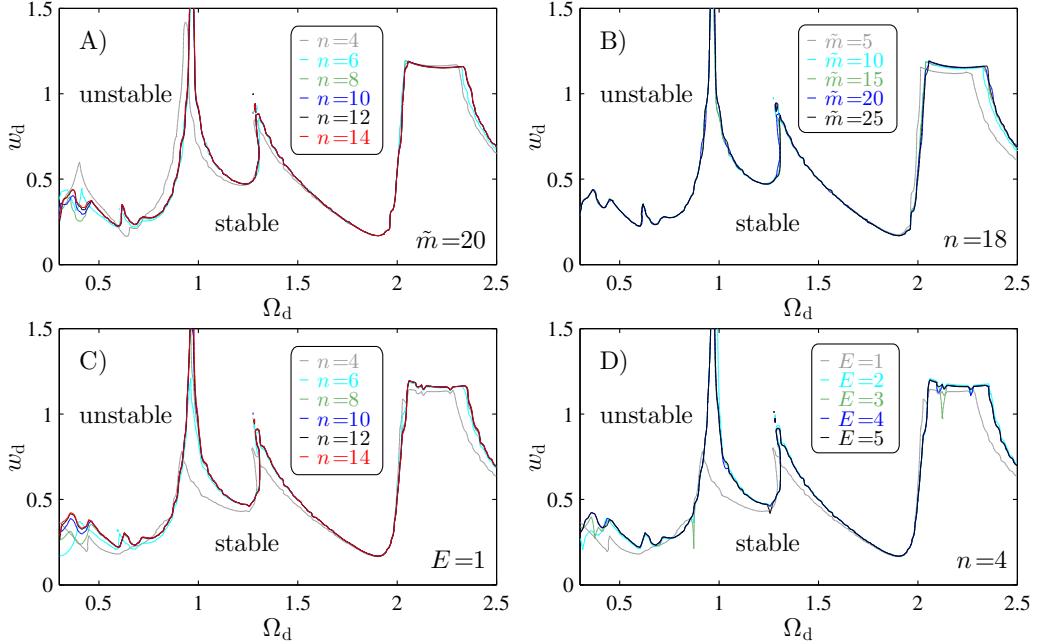


FIGURE 3.13: Stability lobe diagrams for down-milling operations with controlled workpiece holder using parameters $\zeta = 0.05$, $m_r = 10$, $K_r = 3$, $a_e/D = 0.5$, $v = 2$, $q_c = 0.75$, $k_I = 0.1$, $k_P = 0.2$, $k_D = 0.2$, $\delta t = 0.1$ and $\tilde{n} = 4$. Stability diagrams A) and B) were calculated using the PsT method with fixed division number \tilde{m} and order n of polynomial approximation, respectively. Stability diagrams C) and D) were calculated using the SE method with fixed element number E and order n of polynomial approximation, respectively.

The SLD approximations corresponding to (3.37)–(3.38) with (3.42)–(3.43) and (3.47) are shown in Figure 3.13 for a fixed parameter set given in the caption. In panels A) and B), the stability diagrams were computed using the PsT method, while in panels C) and D) the stability diagrams were computed using the SE method. The convergence properties are similar to those in Figure 3.11. Namely, in case of the PsT method, the stability boundaries converge if both polynomial order n and resolution \tilde{m} increase at the same time, while, in case of the SE method, the stability boundaries converge if either polynomial order n or element number E increase.

3.2.3 Milling with controlled tool holder

In Section 3.2.2, the tool holder was fixed while the relative motion between the tool and the workpiece was provided by a digital PID controller. In contrast, this example assumes that the workpiece is fixed while the tool holder's motion is stabilized about a constant v_f feed velocity using a digital PID controller. The corresponding mechanical model is shown in Figure 3.14. Note that this model was first presented

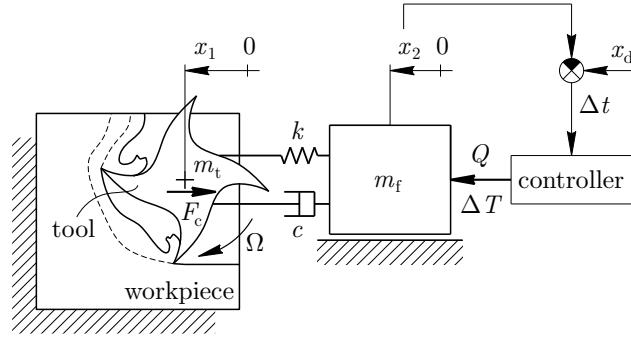


FIGURE 3.14: Model of milling operations with digitally controlled tool holder

in [64] for a different control scheme and without digital effects. The tool holder and the feed drive are modeled by a single block of mass m_f , which is connected to the tool through a spring of stiffness k and a dashpot of viscous damping c . The control force Q is calculated based on the error between the actual position $x_2(t)$ and the desired position $x_d(t) = v_f t$ of the tool holder. The same way as in Section 3.2.1, the time is scaled as $\hat{t} = \omega_n t$, with $\omega_n = \sqrt{k/m_t}$ being the undamped natural angular frequency of the tool. After dropping the hat, the governing equations are again given in the form (3.40)–(3.41), where now

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) - \hat{v}_f t \\ \dot{x}_1(t) - \hat{v}_f \\ x_2(t) - \hat{v}_f t \\ \dot{x}_2(t) - \hat{v}_f \end{bmatrix}, \quad \mathbf{A}_0 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & -2\zeta & 1 & 2\zeta \\ 0 & 0 & 0 & 1 \\ \frac{1}{m_r} & \frac{2\zeta}{m_r} & -\frac{1}{m_r} & -\frac{2\zeta}{m_r} \end{bmatrix}, \quad (3.48)$$

$$\mathbf{f}(t, \mathbf{x}(t), \mathbf{x}(t-\tau_d)) = \frac{F_c(t, \mathbf{x}(t), \mathbf{x}(t-\tau_d))}{m_t \omega_n^2} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad (3.49)$$

with mass ratio $m_r = m_f/m_t$. Here the horizontal component F_c of the resultant cutting force is defined the same way as in Section 3.2.1 except that the difference in the relative position of the tool and the workpiece between two consecutive cuts is changed from (3.44) to

$$\Delta x(\mathbf{x}(t), \mathbf{x}(t-\tau_d)) = x_1(t) - x_1(t-\tau_d). \quad (3.50)$$

The rest of the parameters are given according to Section 3.2.2. The same way as in Section 3.2.2, it is assumed that a periodic stationary solution $(\mathbf{x}_p(t), \mathbf{X}_l^p)$ exists for (3.40)–(3.41). Again, the decomposition of state variables as $(\mathbf{x}(t), \mathbf{X}_l) = (\mathbf{x}_p(t) + \boldsymbol{\xi}(t), \mathbf{X}_l^p + \boldsymbol{\chi}_l)$ and the first-order Taylor expansion of the nonlinear term about $\mathbf{x}_p(t)$ with respect to the perturbation $(\boldsymbol{\xi}(t), \boldsymbol{\chi}_l)$ gives a variational system in the form (3.37)–(3.38), where now matrix \mathbf{A}_0 is defined according to (3.48) and

$$\mathbf{B}(t) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} w_d f_N^{1-q_c} (x_1^p(t) - x_1^p(t-\tau_d))^{q_c-1} \sum_{p=1}^v H_p(t), \quad (3.51)$$

while matrices \mathbf{C} , \mathbf{E} and \mathbf{W}_b are given according to (3.42)–(3.43) and $\xi(t)$ is assembled as (3.45). Similarly to Section 3.2.2, it is assumed that $\tau_d = \tilde{\epsilon}\delta T$ with $\tilde{\epsilon} \in \mathbb{Z}^+$, which simplifies (3.51) to

$$\mathbf{B}(t) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} w_d \sum_{p=1}^v H_p(t). \quad (3.52)$$

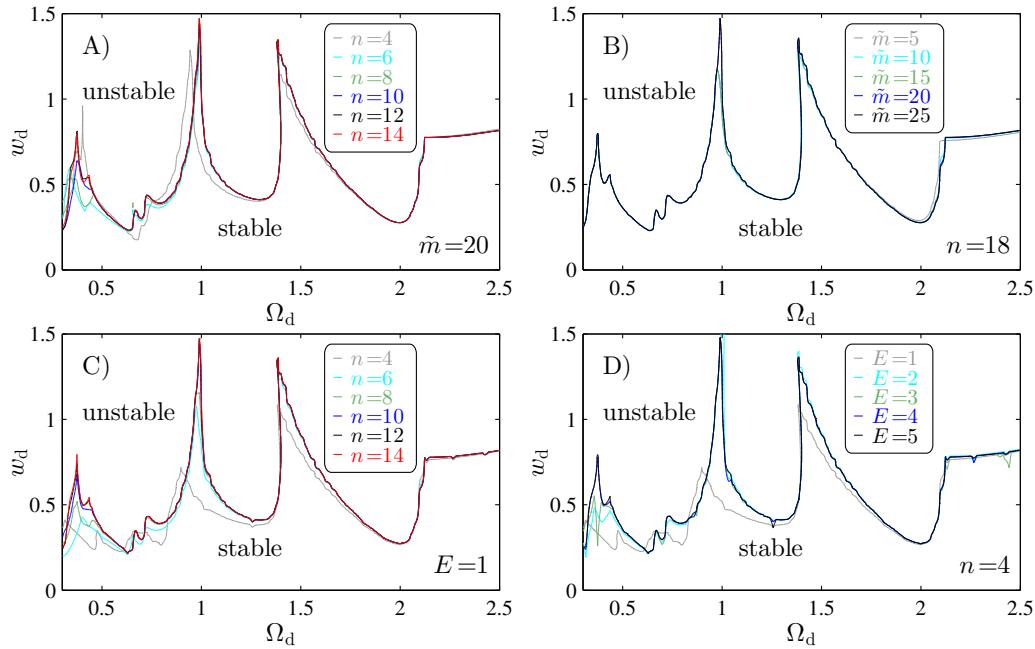


FIGURE 3.15: Stability lobe diagrams for down-milling operations with controlled tool holder using parameters $\zeta = 0.05$, $m_r = 10$, $K_r = 3$, $a_e/D = 0.5$, $v = 2$, $q_c = 0.75$, $k_I = 0.1$, $k_P = 0.2$, $k_D = 0.2$, $\delta t = 0.1$ and $\tilde{n} = 4$. Stability diagrams A) and B) were calculated using the PsT method with fixed division number \tilde{m} and order n of polynomial approximation, respectively. Stability diagrams C) and D) were calculated using the SE method with fixed element number E and order n of polynomial approximation, respectively.

The SLD approximations corresponding to (3.37)–(3.38) with \mathbf{A}_0 given by (3.48), \mathbf{C} , \mathbf{E} and \mathbf{W}_b given by (3.42)–(3.43) and $\mathbf{B}(t)$ given by (3.52) are shown in Figure 3.15 for a fixed parameter set given in the caption. In panels A) and B), the stability diagrams were computed using the PsT method, while in panels C) and D) the stability diagrams were computed using the SE method. The convergence properties are again similar to those of Section 3.2.1–3.2.2: the stability boundaries converge for the PsT method with the increase of both polynomial order n and resolution \tilde{m} , while the stability boundaries for the SE method converge with the increase of either polynomial order n or element number E .

3.3 New results

On the example of the mathematical model of milling operations, I have generalized the spectral element method for the analysis of time-periodic delay-differential equations with discontinuous time-periodic coefficients. The advantage of this generalization is that no adjustment of the element length is necessary in order to guarantee results with exponential convergence rates. The results are summarized as follows.

Thesis 4

Convergent stability boundaries can be achieved by the spectral element method in the stability lobe diagrams of milling operations without the adjustment of the element length. In order to do so, the integral terms of the approximation scheme should be split at the discontinuity fronts of the time-periodic coefficients. The comparison of the spectral element method to the time-domain methods of the machining literature shows that the spectral element method provides converged stability diagrams with smaller computational time. The critical characteristic multipliers of the matrix approximation of the monodromy operator of the governing equations converge faster with respect to the polynomial order, than that of the time-domain methods of the machining literature.

Related publication: [54]

I have analyzed the effect of three different digital feedback control mechanisms on the stability of milling operations. In particular, I have applied a digital control scheme to two existing models of milling processes subjected to feedback control. In addition, I have proposed a new mechanical model, in which the control loop of the drive of the workpiece holder is involved using the same digital control scheme. This control scheme considers the sampling and actuation periods separately and assumes piecewise constant control force. The stability analysis of these systems with fixed sampling period has been carried out for the first time. The results are summarized as follows.

Thesis 5

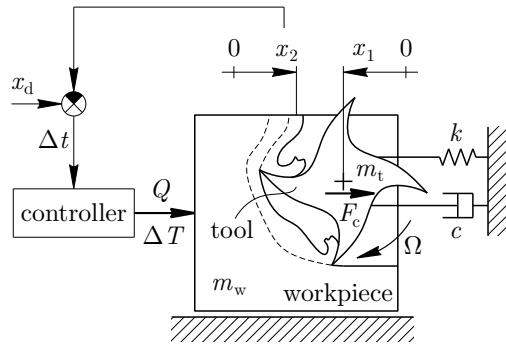


FIGURE 3.16: Model of milling operations with digitally controlled workpiece holder

The mechanical model shown in Figure 3.16 can be used to analyze the effect of the feed drive of the workpiece holder on the stability of milling operations. In the figure m_t , c and k are the modal mass, damping and stiffness of the tool, respectively, whose displacement is measured by x_1 . The spindle speed is Ω , the horizontal cutting force component is F_c , the mass of the workpiece holder together with the workpiece is m_w , whose displacement is measured by x_2 . The control force is Q , the desired position of the workpiece holder is x_d , the sampling period is Δt and the actuation period is $\Delta T = \tilde{n}\Delta t$, where $\tilde{n} \in \mathbb{Z}^+$ is the number of samples per actuation period. With the application of a digital proportional-integral-derivative controller with piecewise constant actuation and numerical integration, the stability lobe diagrams can be determined for this model, for the model of milling operations subjected to active damping and for the model which incorporates the control loop of the feed drive of the tool holder. Depending on the selection of control parameters, these stability diagrams show significant differences from that of the corresponding standard models of the machining literature.

Related publications: [49, 51]

Chapter 4

Stabilizability of delayed systems

The stabilizability of controlled dynamical systems is an important requirement in engineering applications, since many times the maximization of some performance measure of the system is desired while keeping stability. The performance measure is usually defined by a cost function, which is need to be minimized by the proper selection of control parameters. Consequently, the control parameters can be calculated as the solution of a constrained optimization problem. One example for this optimization problem is the selection of the control gains of the active damper in turning processes subjected to active damping, such that the maximum achievable chip width w_{\max} is reached without the occurrence of machine tool chatter. In the following, this optimization problem is studied by assuming digital proportional-derivative (PD) feedback control for the active damper.

In addition to the optimization problem of the active damper in turning, this chapter also deals with the modeling of human balancing tasks, where the balancing processes are described by mechanical systems subjected to proportional-integral-derivative-acceleration (PIDA) delayed feedback control. The loss of balance is studied, which is related to the loss of stabilizability of the closed-loop system. In contrast with optimization problems, in balancing problems of mechanical systems no cost function is needed to be minimized. Instead, the only goal of the controller is to keep the closed-loop system stable about an unstable equilibrium of the open-loop system.

4.1 Stabilizability of turning processes subjected to active damping

One of the most important measures of the performance of manufacturing processes is the material removal rate (MRR), which is the volume of chip removed in a unit time. For turning processes, the nominal MRR is the product of the chip width w , spindle speed Ω and nominal chip thickness h_0 . Since in turning Ω and h_0 are usually limited by the material and shape of the workpiece, major improvements in the MRR can be achieved by the increase of w . However, as it was shown in Chapter 3, w cannot be increased without limits, because of the occurrence of machine tool chatter. Consequently, a reasonable way for the increase of the MRR is to lift the stability boundaries in the stability lobe diagrams (SLDs). These boundaries can be lifted e.g. by building an active damper [65] to the turning machine close to the cutting tool. However, with a fixed Ω , different stability limits can be achieved for w , depending on the selection of control parameters. In the following, a stabilizability diagram is determined for turning processes subjected to active damping, where the active damper is controlled by a digital PD feedback controller. This stabilizability diagram shows the maximum achievable dimensionless specific cutting

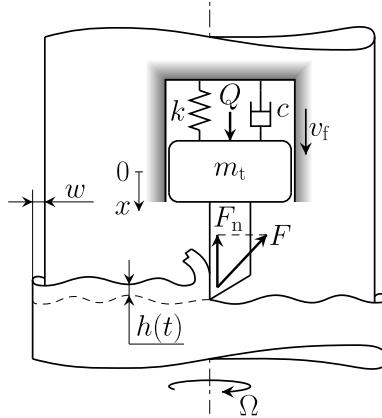


FIGURE 4.1: The mechanical model of turning processes subjected to active damping

force coefficient $w_{d,\max}$ versus the dimensionless spindle speed Ω_d . Since $w_{d,\max}$ is proportional to w_{\max} , the corresponding control gains give maximal MRR.

4.1.1 Modeling

The mechanical model of turning subjected to active damping is shown in Figure 4.1. It is assumed that the active damper is placed close to the cutting tool, therefore it can be considered in the mechanical model as a single force Q acting at the tool tip. Similarly to Section 3.2.1, the tool is modeled by a block of mass m_t which is connected to the tool holder via a spring of stiffness k and a dashpot of viscous damping c . The undamped natural angular frequency of the tool is $\omega_n = \sqrt{k/m_t}$ and the damping ratio is $\zeta = c/(2m_t\omega_n)$. The tool holder is moving with a constant feed velocity v_f with respect to the workpiece which is rotating with angular velocity $2\pi\Omega/60$.

Similarly to Section 3.2.1, the cutting force can be decomposed to normal and tangential components. Here, orthogonal cutting is assumed for which the normal component is parallel to, while the tangential component is perpendicular to the feed velocity. Out of these force components only the normal one plays an important role, since the constraints do not allow the movement of the tool in the tangential direction. Similarly as in (3.33), the normal cutting force component is calculated according to the power law

$$F_n(t) = wK_n h^{q_c}(x(t), x(t - \tau)), \quad (4.1)$$

where now, in contrast with Section 3.2.1, the regenerative delay is $\tau = 60/\Omega$, that is τ is the time of one complete rotation of the workpiece. Chip thickness on the cutting edge

$$h(x(t), x(t - \tau)) = v_f\tau + x(t) - x(t - \tau), \quad (4.2)$$

is calculated as the difference between the tool's current position relative to the workpiece and its relative position one revolution before (see Figure 4.1).

The control force Q is assumed in the same form as in Section 3.2.1, but now with zero integral feedback gain I and with $\tilde{n} = 1$ number of measurements within each actuation period. This latter means that the actuation period coincides with the sampling period, that is $\Delta T = \Delta t$. Consequently, (3.26) with $\mathbf{E} = \mathbf{0}$, $\tilde{n} = 1$ and $F_c(t) = F_n(t)$, becomes the governing equation of the model shown in Figure

4.1, with rescaled time $\hat{t} = \omega_n t$ (the hat is dropped in (3.26)). Vectors \mathbf{f} , \mathbf{x} , \mathbf{x}_d and matrices \mathbf{A}_0 , \mathbf{C} are again given according to (3.28)–(3.30). Dimensionless control parameters k_P , k_D , δt , δT and the dimensionless spindle speed Ω_d are again defined as in Section 3.2.1, while the dimensionless regenerative delay changes to $\tau_d = 2\pi/\Omega_d$.

Similarly as in Section 3.2.1, it is assumed that the stationary (here equilibrium) solution

$$\bar{\mathbf{x}} = \begin{bmatrix} -wK_n(v_f\tau)^{q_c}/k \\ 0 \end{bmatrix} \quad (4.3)$$

of the governing equation is known precisely by the controller and the desired trajectory is set to $\mathbf{x}_d = \bar{\mathbf{x}}$. For perturbation $\xi(t) = \mathbf{x}(t) - \bar{\mathbf{x}}$ about the stationary state, the variational system of the governing equation reads as

$$\dot{\xi}(t) = (\mathbf{A}_0 - \mathbf{B})\xi(t) + \mathbf{B}\xi(t - \tau_d) + \mathbf{C}\xi(t_l - \delta t), \quad t \in [t_l, t_{l+1}), \quad (4.4)$$

with

$$\mathbf{B} = w_d \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}. \quad (4.5)$$

4.1.2 Results

With the stability analysis of (4.4), SLDs can be constructed. In contrast with the previous chapters, the semi-discretization method is used for stability analysis. For (4.4), the matrix approximation \mathbf{U} of monodromy operator $\mathcal{U}(\Xi h)$ is expressed by the semi-discretization method in the form

$$\mathbf{U} = \Phi_{\Xi-1} \Phi_{\Xi-2} \cdots \Phi_0, \quad (4.6)$$

where $\Xi = \text{LCM}(\delta t, \tau_d)$ is the least common multiplier of δt and τ_d , while h is the stepsize of the semi-discretization scheme, $n = \tau_d/h$ is the delay resolution and $\tilde{m} = \delta t/h$ is the period resolution. Matrix $\Phi_u \in \mathbb{R}^{2(n+1) \times 2(n+1)}$ defines a linear map between state vectors \mathbf{X}^u and \mathbf{X}^{u+1} , where

$$\mathbf{X}^u = \left[\tilde{\xi}_{u+1-j} \right]_{j=1}^{n+1} \quad (4.7)$$

approximates the solution segment $\xi_{t_u}(\theta) = \xi(t_u + \theta)$, $\theta \in [-\tau_d, 0]$ at time instant $t_u = uh$ as

$$\xi_{uh}((1-j)h) \approx \tilde{\xi}_{u+1-j}. \quad (4.8)$$

Here $\tilde{\xi}_{u+1-j}$ is given according to $\tilde{\xi}_{k+1} = \tilde{\xi}^k(t_{k+1})$, where $\tilde{\xi}^k(t) = (\tilde{\xi}^k(t), \dot{\tilde{\xi}}^k(t))^T$ is the solution of the ordinary differential equation approximation

$$\begin{aligned} \dot{\tilde{\xi}}^k(t) = & (\mathbf{A}_0 - \mathbf{B})\tilde{\xi}^k(t) + w_d \left(\mathbf{D}_1(t)\tilde{\xi}_{k-n} + \mathbf{D}_2(t)\tilde{\xi}_{k-n+1} \right) \\ & + \mathbf{C}\tilde{\xi}_{(\text{floor}(k, \tilde{m})-1)\tilde{m}}, \quad t \in [t_k, t_{k+1}), \end{aligned} \quad (4.9)$$

of (4.4), with initial condition

$$\tilde{\xi}^k(t_k) = \tilde{\xi}_k. \quad (4.10)$$

In (4.9), matrices $\mathbf{D}_1, \mathbf{D}_2 \in \mathbb{R}^{2 \times 2}$ are defined such that

$$p_3(t) = [0, 1] \left(\mathbf{D}_1(t) \tilde{\boldsymbol{\xi}}_{k-n} + \mathbf{D}_2(t) \tilde{\boldsymbol{\xi}}_{k-n+1} \right) \quad (4.11)$$

gives a third order Hermite polynomial approximation for $\tilde{\xi}^k(t - \tau_d)$ in time interval $t \in [t_k, t_{k+1})$. Finally, the elements of Φ_k matrices can be calculated by solving (4.9)–(4.10) for $k = 0, \dots, \Xi - 1$. Details on the calculation of $\mathbf{D}_1(t)$, $\mathbf{D}_2(t)$ and Φ_k matrices can be found in [47].

Note that $\text{LCM}(\delta t, \tau_d) = \text{LCM}(\tilde{m}, n)$, thus while the size of \mathbf{U} is determined only by the delay resolution n , the number of matrix multiplications in (4.6) depends on the ratio of n and period resolution \tilde{m} . This latter creates difficulty during the calculation of SLDs due to that the dimensionless sampling time δt is fixed (it is given by the sampling frequency f_s of the controller), while τ_d is inversely proportional to Ω_d . For accurate results, n should be kept high enough: according to [47] $n \geq 30$ is necessary for good approximation. At the same time, for the achievement of a closely equidistant grid on the Ω_d axis, $\Omega_d = 2\pi/(nh)$ has to be close to the ideal grid. However, since δt is fixed, n cannot be chosen arbitrarily close to the ideal grid because $\Xi = \text{LCM}(\delta t, \tau_d)$ has to be kept low. Consequently, choosing the gridpoints along the Ω_d axis needs special care in order to keep reasonable computation times.

The stability of (4.4) depends on machining parameters Ω_d , w_d , system parameter ζ and control parameters δt , k_P and k_D . For the semi-discretized system with fixed system parameter ζ , the stabilizability problem is defined by the objective function

$$J_{\text{obj}} = -w_{d,\text{crit}}(\Omega_d, \delta t, k_P, k_D), \quad (4.12)$$

where $w_{d,\text{crit}}$ is the stability limit of the matrix approximation (4.6) of the monodromy operator. Objective function (4.12) is to be minimized.

For fixed δt and Ω_d , the limit $w_{d,\text{max}}$, above which stabilization is not possible with any (k_P, k_D) control parameter combination can be calculated by increasing w_d and tracing the disappearance of stable domain in the (k_P, k_D) plane of control parameters. By repeating this calculation on several gridpoints of the Ω_d axis, a stabilizability diagram can be computed which shows $w_{d,\text{max}}$ and the corresponding $k_{P,\text{max}}, k_{D,\text{max}}$ control gains versus Ω_d . The concept of the calculation of stabilizability limits is shown in Figure 4.2, where a series of stability diagrams is drawn in the plane (k_P, k_D) for different dimensionless spindle speeds Ω_d and dimensionless depth of cuts w_d . Clearly, the stabilizability boundary is between the two gridpoints of w_d , where the stable domain disappears in the corresponding stability diagrams. With the refinement of the number of diagrams in this region of the w_d axis, a more accurate location of the limit of stabilizability can be given. Below, the steps of the computation of the stabilizability diagram is provided in more detail.

1. The system parameter ζ and dimensionless sampling period δt are fixed.
2. The investigated region of the dimensionless spindle speed $[\Omega_{d,I}, \Omega_{d,II}]$ is divided onto N number of non-equidistant intervals by discrete values $\Omega_{d,i}$, $i = 1, 2, \dots, N$ such that the ratio of the corresponding dimensionless regenerative delay $\tau_{d,i} = 2\pi/\Omega_{d,i}$ and the dimensionless sampling period δt is a rational number, i.e., $\tau_{d,i}/\delta t = n_i/\tilde{m}_i$, where $n_i, \tilde{m}_i \in \mathbb{Z}^+$ give the delay and the period resolutions, respectively. The resolution of the axis Ω_d is such that $\Omega_{d,i+1} - \Omega_{d,i} < \frac{2}{N}(\Omega_{d,II} - \Omega_{d,I})$ and $n_i \geq 30$ for all $i = 1, 2, \dots, N$.

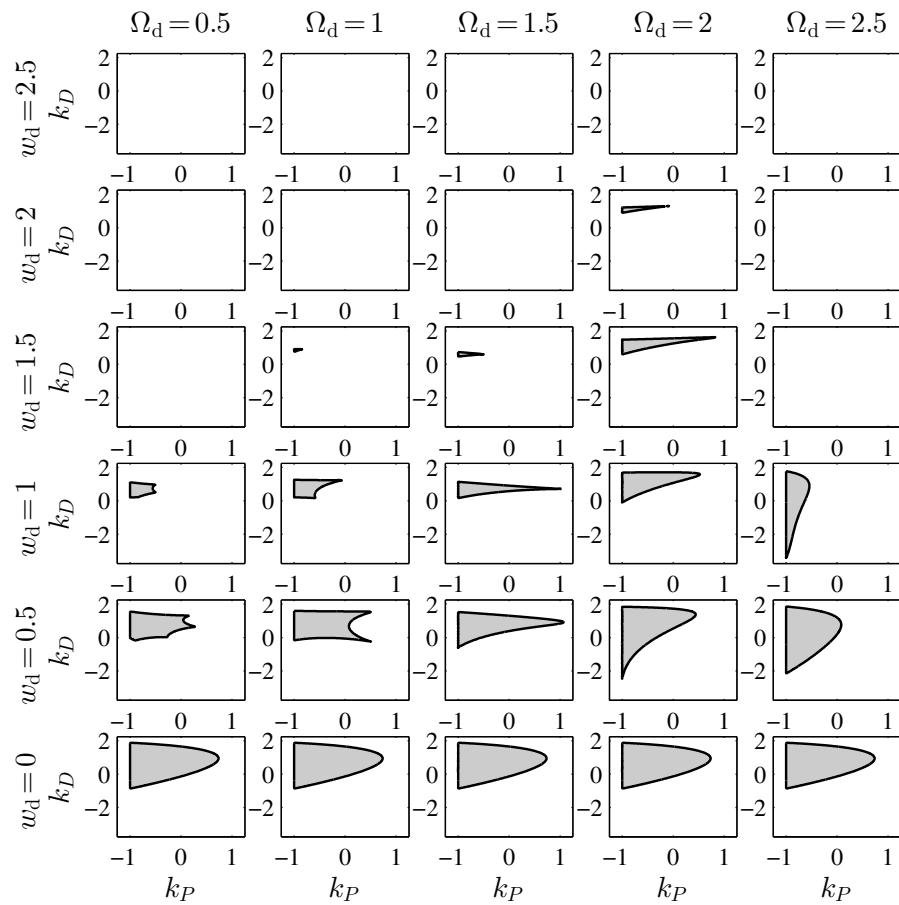


FIGURE 4.2: Stability diagrams on the plane of (k_P, k_D) dimensionless control gains for $\delta t = 0.5$ dimensionless sampling time, $\zeta = 0.05$ damping ratio and different Ω_d and w_d values

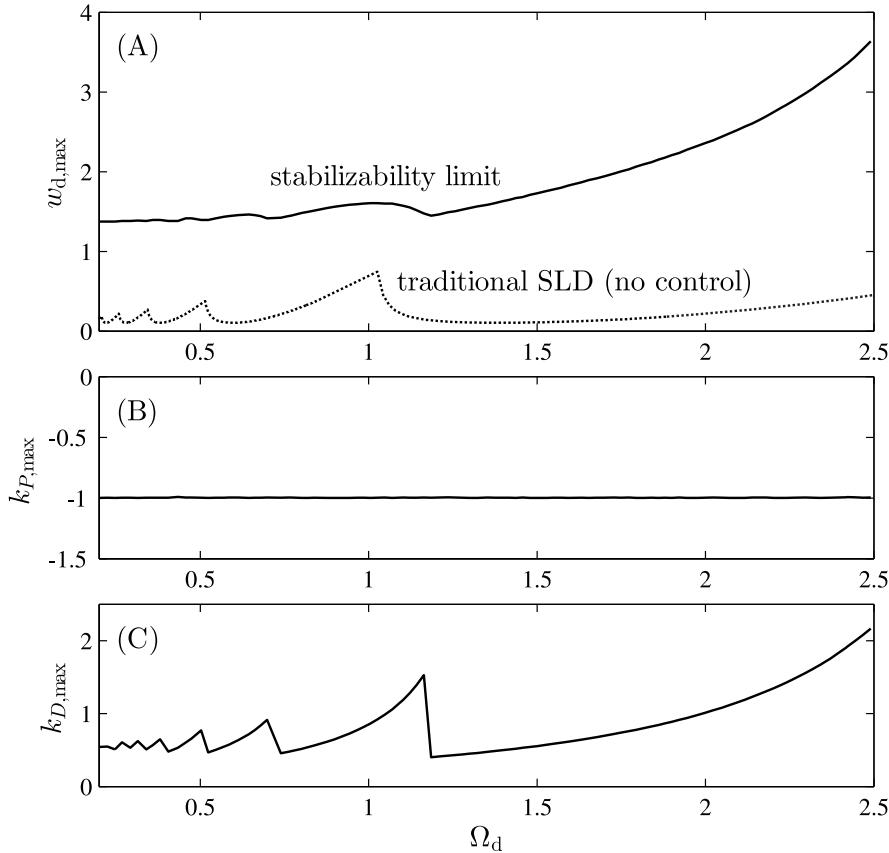


FIGURE 4.3: Stabilizability diagram of turning processes with damping ratio $\zeta = 0.05$, subjected to active damping using digital PD feedback controller with sampling frequency $f_s = 2\omega_n$ ($\delta t = 0.5$); A) the maximum applicable dimensionless specific cutting force coefficient $w_{d,\max}$ versus the dimensionless spindle speed Ω_d ; B) the dimensionless proportional gain, associated with $w_{d,\max}$; C) the dimensionless derivative gain associated with $w_{d,\max}$

3. The spindle speed is fixed step-by-step to the discrete values $\Omega_d = \Omega_{d,i}$, $i = 1, 2, \dots, N$. For each fixed $\Omega_d = \Omega_{d,i}$, the stepsize of the semi-discretization scheme is determined as $h_i = \tau_{d,i}/n_i$.
4. The investigated region of the dimensionless depth of cut $[w_{d,I}, w_{d,II}]$ is divided onto $(M - 1)$ equidistant initial intervals by discrete values $w_{d,j} = j\Delta w_d$, $j = 1, 2, \dots, M$ with $\Delta w_d = (w_{d,II} - w_{d,I})/M$. The stability diagrams in the plane (k_P, k_D) are determined for all $w_d = w_{d,j}$, $j = 1, 2, \dots, M$. If the case $w_d = w_{d,j}$ can be stabilized but there is no stable domain for $w_d = w_{d,j+1}$, then the interval $[w_{d,j}, w_{d,j+1}]$ is further investigated by interval halving method, i.e., stability diagrams are determined for $w_d = w_{d,j} + \frac{1}{2}\Delta w_d$, then either for $w_d = w_{d,j} + \frac{1}{4}\Delta w_d$ or for $w_d = w_{d,j} + \frac{3}{4}\Delta w_d$, etc. The procedure is stopped if w_d is given with an accuracy of $\Delta w_d/2^7$ and its value is recorded as $w_{d,\max}$.
5. The maximum value $w_{d,\max}$ is determined for all discrete spindle speeds $\Omega_{d,i}$, $i = 1, 2, \dots, N$ and the stabilizability diagram is plotted.

Figure 4.3 presents the stabilizability diagram for the dimensionless sampling period $\delta t = 0.5$ and damping ratio $\zeta = 0.05$. The boundaries of the investigated parameter domain are $[\Omega_{d,I}, \Omega_{d,II}] = [0.2, 2.5]$ and $[w_{d,I}, w_{d,II}] = [0, 4]$ with $N = 100$,

$M = 4$. The actual sampling period is $\Delta t = \delta t/\omega_n = 0.5/\omega_n$, i.e., the digital control samples the state at a frequency $f_s = 1/\Delta t = 2\omega_n$, which is the double of the natural frequency of the system. The traditional SLD (i.e., the uncontrolled case, see [26]) is also shown by dotted line. It can be seen that the depth of cut can significantly be increased. At the resonant spindle speed $\Omega_d = 1/k$, $k = 1, 2, \dots$ the maximum dimensionless depth of cut is about the double of the traditional SLD, while in between the resonant spindle speeds, the depth of cut is increased by a factor of 8~10. Note that this improvement is achieved in spite of the fact that the sampling frequency of the controller is relatively low compared to natural frequency of the uncontrolled system ($f = 2\omega_n$). Figure 4.3 also presents the dimensionless proportional and derivative control gains, which are associated with the disappearance of the stable domain in the plane (k_P, k_D) . Note that $k_{P,\max} = -1$, which means that the stable domain disappears at the stability limit $k_P = -1$ as it can also be observed in Figure 4.2.

The above presented analysis is based on an ideal model, since it does not consider any noise, modeling uncertainties, and the parameters of the machining system are also perfectly known at all time instances. In practice, the measured output is affected by noise and the system parameters are changing during the operation due to changing machining conditions (e.g., tool wear, changes in the local temperature on the active face of the tool, etc.). Intuitively, if the uncertainties in modeling, in system parameters and in the output were also taken into account then the gain in the maximum depth of cut would decrease. In spite of these modeling restrictions, the current analysis still presents a general view on the stabilizability of machining operation. Furthermore, it was shown that the interplay between the regenerative delay and the digital effects of the controller may strongly affect the stabilizability of the system.

4.2 The modeling of human balancing with PIDA control

Human balancing is one of the most significant tasks of human's everyday movements. In the mechanical point of view, balancing means the stabilization of an object about its equilibrium. In human balancing, the process of stabilization is controlled by the brain: the brain carries out "measurements" with sensory organs and based on these measurements it gives commands to the musculature. However, there is a time-gap between the detection and the realization of the commands given to the musculature. The main cause of this time-delay (or reaction time) is that the processing of the received information, decision making and its realization require time. Similar stabilization problems show up in many industrial applications, where an unstable process is stabilized by feedback controllers. Controllers carry the traits of human balancing since they take measurements on the mechanical system and actuate according to algorithms based on the measured data.

The question arises: what is the algorithm which determines the command to the musculature based on the information provided by the sensory organs of humans? The literature of biomechanics has been working on this question for several decades [34, 35, 63, 90]. In the following, a possible operation of the human brain during the balancing process is investigated by the modeling and analysis of two balancing tasks: stick balancing [14, 42, 58] and postural balancing [4, 28, 57, 89].

First, the mechanical models of the two balancing tasks are constructed, where the human interaction is considered by a single force or torque vector. This force or torque is assumed to be governed by a PIDA delayed-feedback controller, where

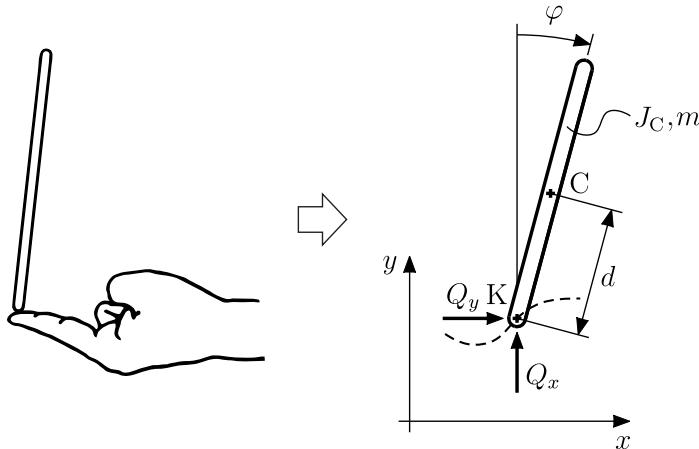


FIGURE 4.4: The mechanical model of stick balancing

the delay models the time-gap between movement detection and realization of muscle movements. The proportional, derivative and acceleration feedback concept assumes that the position, velocity and the acceleration of the balanced object are all measured by the human sensory system. When these measured state values do not coincide with the desired ones, a corrective force is applied on the balanced object by the human, which is proportional to the error between the measured and the desired state. Furthermore, the integral gain takes into account that during the fine positioning of lifted items the static error resulting from modeling uncertainties and friction has to be eliminated. In the industry, this static error is typically eliminated by the inclusion of the integral of the position error in the feedback loop. This gave the idea to add an integral gain to the control model of [28], where the process of human balancing is considered by a proportional-derivative-acceleration feedback controller.

After the construction of the mathematical models of the two balancing tasks, their stability is analyzed around their equilibrium using locally linearized equations. The loss of balance is assumed to occur when the locally linearized system cannot be stabilized by any control parameter set, that is when the locally linearized system is not stabilizable. Consequently, after the construction of the mathematical models their stabilizability is analyzed for small perturbations about their equilibria. Finally, the results of the stabilizability analysis is presented and discussed for the two balancing tasks based on experimental data.

4.2.1 Balancing models

Stick balancing is modeled by a bar and a single control force vector. It is assumed that the air resistance can be neglected and the friction on the finger is large enough to maintain zero relative velocity between the contact point of the finger and the bar throughout balancing. The 3 DoF mechanical model of stick balancing is shown in Figure 4.4. The displacement of contact point K is measured by coordinates x, y , while φ stands for the angular position of the bar. The governing

equations of the mechanical model are

$$m\ddot{x}(t) + md\ddot{\varphi}(t) \cos(\varphi(t)) - md\dot{\varphi}^2(t) \sin(\varphi(t)) = Q_x(t), \quad (4.13)$$

$$m\ddot{y}(t) - md\ddot{\varphi}(t) \sin(\varphi(t)) - md\dot{\varphi}^2(t) \cos(\varphi(t)) = Q_y(t) - mg, \quad (4.14)$$

$$J_C\ddot{\varphi}(t) = Q_y(t)d \sin(\varphi(t)) - Q_x(t)d \cos(\varphi(t)), \quad (4.15)$$

where m is the mass of the bar and J_C is its mass moment of inertia about its centroid C. The distance of the centroid of the bar from the contact point is denoted by d , while g is the gravitational acceleration. The interaction between the finger and the stick is modeled by a single force vector $\mathbf{Q}(t) = (Q_x(t), Q_y(t))^T$. This force vector can be decomposed to feedforward and feedback terms as $\mathbf{Q}(t) = \mathbf{Q}^{ff}(t) + \mathbf{Q}^{fb}(t)$. The feedforward term $\mathbf{Q}^{ff}(t)$ is determined such that it moves the system along a desired trajectory $(x_d(t), y_d(t), \varphi_d(t))$. The goal of stick balancing is to keep the stick in a vertical position, therefore the desired trajectory is $(x_d(t), y_d(t), \varphi_d(t)) \equiv (0, 0, 0)$ for which the corresponding feedforward term is $\mathbf{Q}^{ff}(t) \equiv (0, mg)^T$. During stick balancing the goal is to maintain the stick in the upright position, therefore the feedback control force is assumed to depend only on the angular position of the stick. Furthermore, since a vertical movement of the bottom point of the stick changes the angular position less than its movement in the horizontal direction and since in reality humans typically use the vertical movement only for "rescue actions" during stick balancing, the vertical component of the feedback force is assumed to be zero. Consequently, the feedback force vector obtains the form $\mathbf{Q}^{fb}(t) = (Q_x^{fb}(t), 0)^T$, where the horizontal feedback force component is assumed to be given according to a PIDA control rule of the form

$$Q_x^{fb}(t) = I \int_{-\infty}^t \varphi(\theta - \tau) d\theta + P\varphi(t - \tau) + D\dot{\varphi}(t - \tau) + A\ddot{\varphi}(t - \tau), \quad (4.16)$$

where I , P , D and A are the integral, proportional, derivative and acceleration feedback gains, respectively. The reaction time of humans is modeled in the control rule by time delay τ . After the linearization of (4.13)–(4.15) about the equilibrium state $(x, \dot{x}, y, \dot{y}, \varphi, \dot{\varphi}) = (0, 0, 0, 0, 0, 0)$, one obtains

$$m\ddot{x}(t) + md\ddot{\varphi}(t) = Q_x^{fb}(t), \quad (4.17)$$

$$m\ddot{y}(t) = 0, \quad (4.18)$$

$$J_C\ddot{\varphi}(t) - mgd\varphi(t) = -Q_x^{fb}(t)d. \quad (4.19)$$

Note, however, that above it was assumed that parameters m , J_C and d are known exactly. If the controller does not know the exact value of these parameters, then the feedforward force will be inaccurate, thus the linearized equations (4.17)–(4.19) will obtain a different form. In the following, the exact values of all the system parameters are assumed to be precisely known by the controller. Furthermore, it is also assumed that the stick is prismatic and homogeneous, which results in $d = l/2$ and $J_C = ml^2/12$ with l being the length of the stick.

Note that by using (4.16), (4.19) contains only one state variable: φ . Consequently, (4.19) can be detached from the system (4.17)–(4.19) and it can be solved separately. As a result, in (4.17), the terms containing φ can be considered as an excitation with explicit time-dependency. Equation (4.18) is also separated from (4.17)–(4.19), in fact there is no motion in direction y . It can be thus concluded that the stability of (4.17)–(4.19) is determined solely by (4.19). With the introduction of

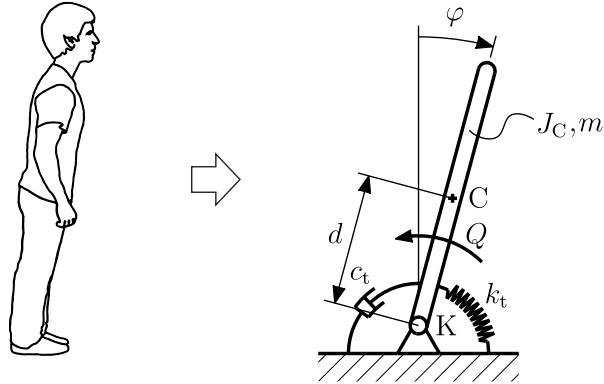


FIGURE 4.5: The mechanical model of postural sway

dimensionless time $\hat{t} = t/\tau$ and dropping the hat immediately, (4.19) reads

$$\ddot{\varphi}(t) - a\varphi(t) = -k_I \int_{-\infty}^t \varphi(\theta - 1) d\theta - k_P\varphi(t - 1) - k_D\dot{\varphi}(t - 1) - k_A\ddot{\varphi}(t - 1), \quad (4.20)$$

where $a = 6g\tau^2/l$ is a dimensionless system parameter, while $k_I = 6I\tau^3/(ml)$, $k_P = 6P\tau^2/(ml)$, $k_D = 6D\tau/(ml)$, $k_A = 6A/(ml)$ are the dimensionless integral, proportional, derivative and acceleration gains, respectively.

Postural sway, also called quiet stance, is the process of human balancing while standing in one place. The mechanical model of postural sway is shown in Figure 4.5, which considers the human body by a prismatic bar. The contact between the feet and the ground is modeled by a fixed joint at contact point K. The resistance of the human ankle against rotation is modeled by a linear spring of torsional stiffness k_t and dashpot of torsional viscous damping c_t . Again, the distance between the contact point K and centroid C is denoted by d , the mass of the human body is m and its mass moment of inertia about centroid C is J_C . The governing equation of the mechanical model is

$$J_K\ddot{\varphi}(t) + c_t\dot{\varphi}(t) + k_t\varphi(t) - mgd\sin(\varphi(t)) = -Q(t), \quad (4.21)$$

where $J_K = J_C + md^2$ is the mass moment of inertia about contact point K. Similarly as in case of stick balancing, the control torque $Q(t) = Q^{ff}(t) + Q^{fb}(t)$ is split onto feedforward and to feedback terms. The desired trajectory of the balancing task is $\varphi_d(t) = 0$, for which the corresponding feedforward term is $Q^{ff}(t) \equiv 0$. The feedback term $Q^{fb}(t)$ of the control torque is again defined as $Q_x^{fb}(t)$ in (4.16). After the linearization of (4.21) about the desired trajectory and the introduction of dimensionless time $\hat{t} = t/\tau$ while dropping the tilde immediately, one arrives at

$$\ddot{\varphi}(t) + b\dot{\varphi}(t) - a\varphi(t) = -k_I \int_{-\infty}^t \varphi(\theta - 1) d\theta - k_P\varphi(t - 1) - k_D\dot{\varphi}(t - 1) - k_A\ddot{\varphi}(t - 1), \quad (4.22)$$

where the system parameters are now $a = (mgd - k_t)\tau^2/J_K$ and $b = c_t\tau/J_K$, while the dimensionless control parameters k_I, k_P, k_D, k_A are defined the same way as for stick balancing. Note that formally (4.22) incorporates (4.20) (with $b = 0$), therefore, in the sequel, the stabilizability analysis is detailed only for (4.22) with general parameter sets.

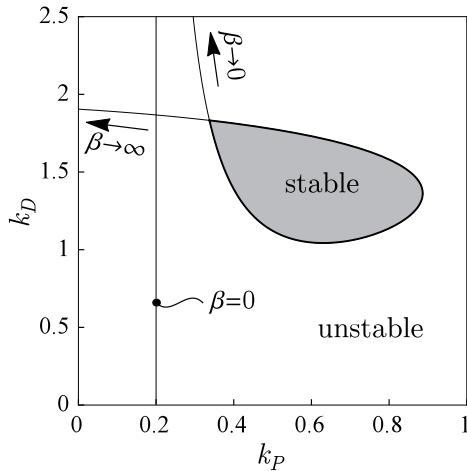


FIGURE 4.6: Stable parameter domain for system parameters $a = 0.2$, $b = 0$ and control parameters $k_I = 0.2$, $k_A = 0.2$

4.2.2 Stabilizability analysis

In the following analysis, it is assumed that system parameter b has a fixed value in (4.22). The critical system parameter $a = a_{\text{cr}}$ is sought above which (4.22) is not stabilizable, that is, above which (4.22) is unstable for any (k_I, k_P, k_D, k_A) control parameter set.

For the stability analysis of (4.22), the D-subdivision method [75] is used which was briefly detailed in Section 2.3.1. The D-curves correspond to the characteristic roots $\lambda = \alpha + i\beta$ with real part $\alpha = 0$, that is to the characteristic roots which are located at the imaginary axis. The D-curves can be expressed from the characteristic equation of (4.22) and they are given by

$$\beta = 0 : \begin{cases} k_P = a, \\ k_D \in \mathbb{R}, \end{cases} \quad (4.23)$$

$$\beta \neq 0 : \begin{cases} k_P = (a + \beta^2) \cos(\beta) + b\beta \sin(\beta) + k_A \beta^2, \\ k_D = \frac{a + \beta^2}{\beta} \sin(\beta) - b \cos(\beta) + \frac{k_I}{\beta^2}, \end{cases} \quad (4.24)$$

where $\beta \in [0, \infty)$ is a running parameter (the frequency of the characteristic root). For fixed a , k_I and k_A , these $(k_P(\beta), k_D(\beta))$ parametric curves split the (k_P, k_D) parameter plane onto domains in which the number of unstable characteristic roots is constant. Those domains are sought which have zero number of unstable characteristic roots. These domains can be found e.g. using numerical methods or the so-called Stepan formulas (see Theorem 2.19. in [75]). Such stable domain is shown in Figure 4.6 for fixed a , k_I and k_A parameters. In the parameter plane (k_P, k_D) , the stability boundary is drawn by thick and the D-curves are drawn by thin lines, while the stable domain is shown by grey color. Panels A) and B) of Figure 4.7 and panel C) of Figure 4.8 show the stable domains in the parameter plane (k_P, k_D) encircled by thick lines. These panels illustrate the typical behavior of the stable domain under the increase of k_I , a and k_A , respectively. It can be seen, that the stable domain shrinks with the increase of a or k_I and expands with the increase of k_A . For each (k_I, k_A) parameter combination, an a_{cr} critical system parameter can be found above which the stable domain disappears in the (k_P, k_D) plane. This

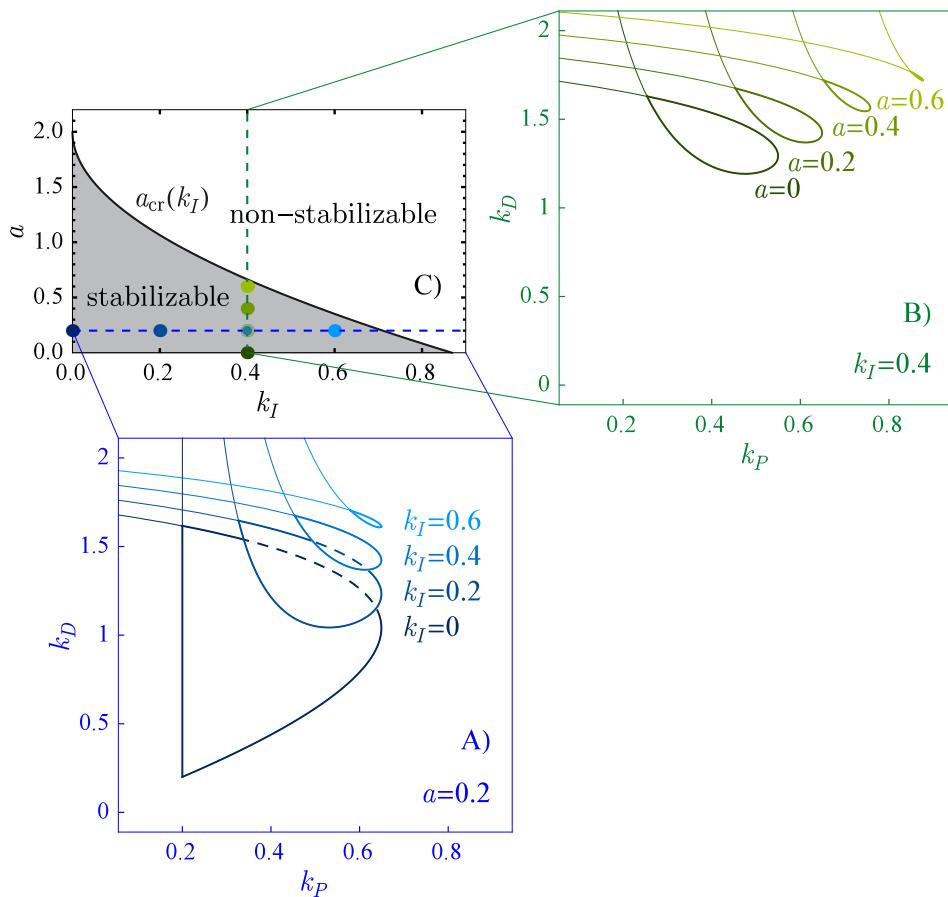


FIGURE 4.7: Stabilizability diagram of (4.22) for $b = 0$ and $k_A = 0$: A) stable parameter domain with fixed $a = 0.2$ and increasing k_I values; B) stable parameter domain with fixed $k_I = 0.4$ and increasing a values; C) $a_{cr}(k_I)$ critical system parameter as a function of k_I

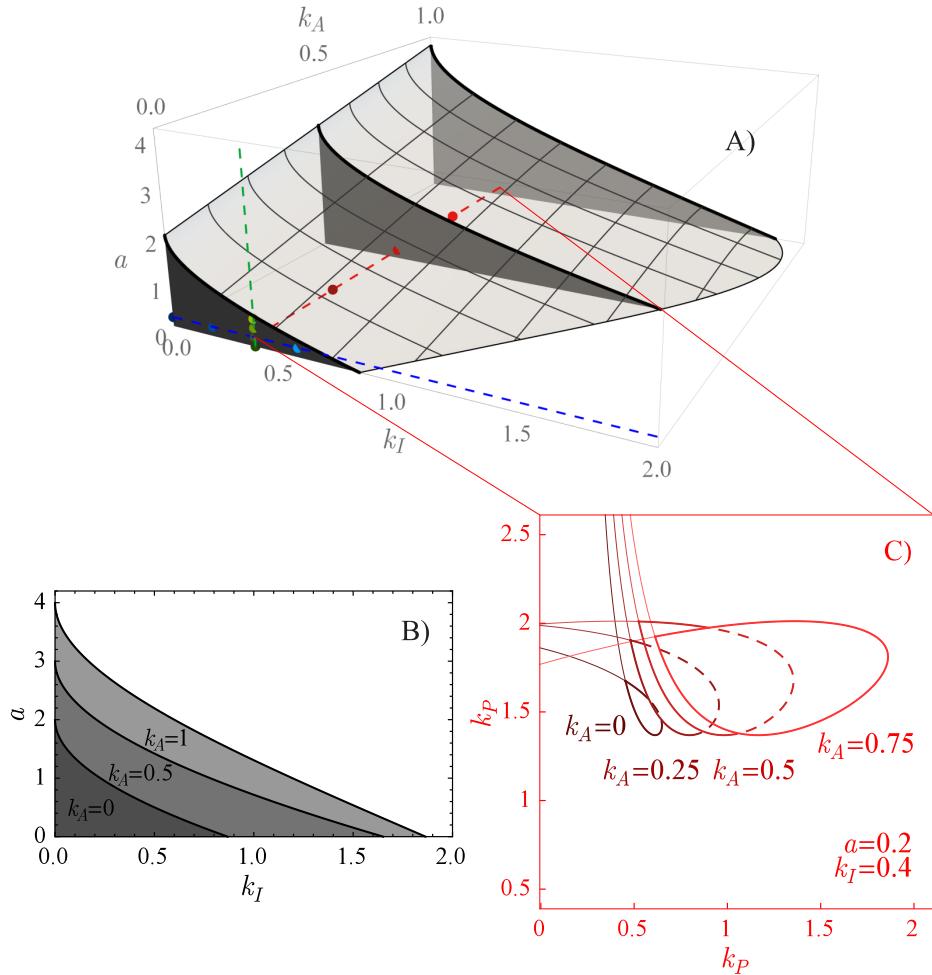


FIGURE 4.8: Stabilizability diagram of (4.22) for $b = 0$: A) $a_{\text{cr}}(k_I, k_A)$ critical system parameter as a function of k_I and k_A ; B) stabilizable domain with increasing k_A ; C) stable parameter domain with fixed $a = 0.2$, $k_I = 0.4$ and increasing k_A values

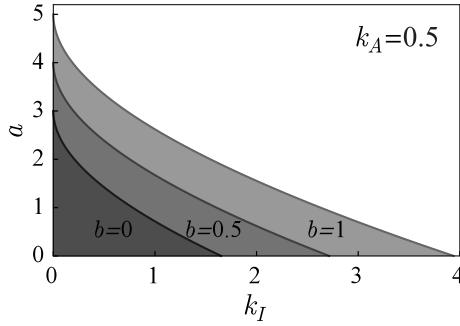


FIGURE 4.9: Stabilizable domain of (4.22) for $k_A = 0.5$, with increasing b values

means, that if $a > a_{\text{cr}}(k_I, k_A)$, then (4.22) cannot be stabilized for any (k_P, k_D) parameter set. As Figure 4.6 illustrates, functions $k_P(\beta)$ and $k_D(\beta)$ (which describe the D-curves) have one-one local extremum in the investigated part of the (k_P, k_D) plane. The disappearance of the stable domain occurs when the kink of the D-curve closes that is when these local extrema coincide. Consequently, the condition of the disappearance of the stable domain is

$$\frac{dk_P}{d\beta}(a_{\text{cr}}, \beta_{\text{cr}}) = 2k_A\beta_{\text{cr}} + (2+b)\beta_{\text{cr}} \cos(\beta_{\text{cr}}) - (a_{\text{cr}} - b + \beta_{\text{cr}}^2) \sin(\beta_{\text{cr}}) = 0, \quad (4.25)$$

$$\frac{dk_D}{d\beta}(a_{\text{cr}}, \beta_{\text{cr}}) = -\frac{2k_I}{\beta_{\text{cr}}^3} + \frac{(1+b)\beta_{\text{cr}}^2 - a_{\text{cr}}}{\beta_{\text{cr}}^2} \sin(\beta_{\text{cr}}) + \frac{a_{\text{cr}} + \beta_{\text{cr}}^2}{\beta_{\text{cr}}} \cos(\beta_{\text{cr}}) = 0, \quad (4.26)$$

which gives a system of nonlinear equations for a_{cr} and β_{cr} . This system of equations is solved using the built-in nonlinear equation solver of the software Wolfram Mathematica, which employs a Newton-Raphson scheme. With the calculation of a_{cr} over a grid of parameter plane (k_I, k_A) , the two-parameter function $a_{\text{cr}}(k_I, k_A)$ can be determined. For the nonlinear solver, the close-enough initial guess at each gridpoint was produced by the result at a neighboring gridpoint, while the initial guess at the first gridpoint was calculated using the D-curve plots on the (k_P, k_D) parameter plane close to the disappearance of the kink. Below $a_{\text{cr}}(k_I, k_A)$, (4.22) is stabilizable, hence the stabilizable sets of (a, k_I, k_A) can be visualized in a 3-dimensional space, as it is shown in panel A) of Figure 4.8, where the gray volume illustrates the domain of stabilizable parameter sets. Corresponding to fixed k_A values, different sections of this volume are visualized by darker gray colors in panels A) and B) of Figure 4.8 and panel C) of Figure 4.7.

At the beginning of Section 4.2.2, it was assumed that system parameter b has a fixed value. In Figures 4.6–4.8, $b = 0$ was used. In order to show the effect of b on the stabilizable domain of parameter plane (k_I, a) the stabilizability boundaries of (4.22) are computed for fixed k_A and increasing b parameter values. The results are shown in Figure 4.9. It can be seen that the stabilizable domain increases with the increase of the damping term b .

4.2.3 Results

In the following, the above determined results for the stabilizability of (4.22) are compared with experimental results given in the literature.

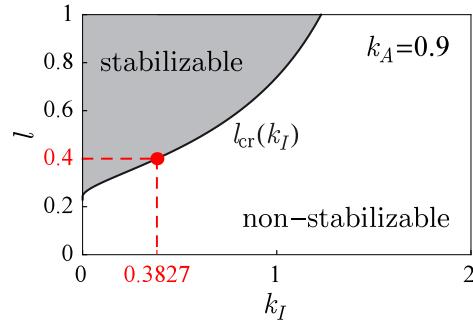


FIGURE 4.10: Stabilizability diagram of stick balancing on the (k_I, l) parameter plane of dimensionless integral feedback gain and stick length with reflex delay $\tau = 125$ [ms] and dimensionless acceleration feedback gain $k_A = 0.9$

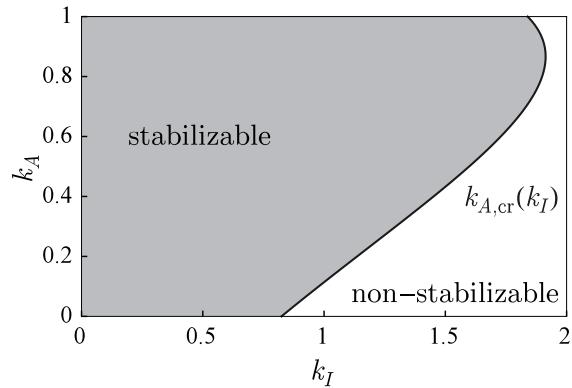
Stick balancing is investigated in [14, 58]. Based on physiological experiments, these papers estimated the reflex delay of human visuomotor control to be in the range $\tau \in [100, 250]$ [ms]. By choosing the reflex delay of balancing as $\tau = 125$ [ms] and the dimensionless acceleration feedback gain as $k_A = 0.9$, the above derived results for the stabilizability of (4.22) give Figure 4.10, where the increase of the integral gain k_I does not decrease the critical length l_{crit} of the stick. In fact, as panel B) of Figure 4.8 shows, a_{cr} is monotonously decreasing with the increase of k_I and since a is inversely proportional to l , this means that k_I cannot decrease the critical length of the stick for any control or system parameters. As a result, the integral gain k_I cannot improve the balancing performance of the model.

According to [15], the minimum length of the stick which can be balanced by humans is about $l = 40$ [cm]. As it is shown in Figure 4.10, the dimensionless integral gain corresponding to the critical length $l_{\text{crit}} = 40$ [cm] is $k_I = 0.3827$. Therefore, the above presented PIDA control rule can be used to explain the stick balancing process. However, many important aspects of human balancing have not been addressed by the analyzed model such as the dead zones of movement detection, parameter uncertainties in the model used by the controller and the uncertainties which are present in the state variables measured by the sensory organs of humans. These unmodeled aspects may lead to the loss of stability and therefore to the increase in the calculated critical stick length l_{crit} . It is also important to mention that several other control rules exist in the literature which also result in critical length $l = 40$ [cm] (see [27]). As a result, the validity of the above presented model needs further investigation.

Postural sway is the subject of [4]. Based on the results of this paper, the mechanical parameters in (4.21) are chosen according to Table 4.1, which result in system parameters $a = 0.078$ and $b = 0.013$. Using these parameters, the above derived results for the stabilizability of (4.22) give Figure 4.11. It can be seen that the parameter combination of k_I and k_A can be chosen from a broad domain without the loss of stabilizability. Consequently, the experimental results are justified for a wide range of control parameters: the patient whose mechanical parameters are given in Table 4.1 is able to keep balance. Furthermore, the wide range for the selection of the stabilizing combination of control parameters k_I and k_A reflects that the balancing task is easy, which matches with the every-day experience of healthy humans. However, it is important to mention again that several other control rules can be

m	60 kg
d	1 m
J_K	60 kgm^2
k_t	471 Nm/rad
c_t	4.0 Nms/rad
τ	0.2 s

TABLE 4.1: Mechanical and physiological parameters of (4.21) according to [4]

FIGURE 4.11: Stabilizability diagram of postural sway on the (k_I, k_A) parameter plane of dimensionless integral and acceleration feedback gains with dimensionless system parameters $a = 0.078$ and $b = 0.013$

found in the literature which give similar results, therefore the further analysis of the model and the comparison of results with experiments are necessary.

4.3 New results

I have investigated the effect of active damper on the stability and stabilizability of turning operations. The active damper was assumed to be controlled by a digital proportional-derivative feedback controller, which generates a piecewise constant force acting on the tool. The results are summarized as follows.

Thesis 6

In the mechanical model of turning operations subjected to active damping, the material removal rate can be increased by the proper tuning of the control gains of the active damper. The proper tuning of the control parameters can be carried out using stability diagrams. In case of an active damper controlled by a digital proportional-derivative feedback loop with piecewise constant control force, the omission of the delay, caused by sampling, and the piecewise constant nature of the control force can lead to significant differences in the stability diagrams. The sampling effect of the digital controller limits the maximum achievable material removal rate. This limitation is captured by the stabilizability diagram which shows the maximum achievable specific cutting force coefficient versus the spindle speed.

Related publications: [46, 47, 52, 53]

I have modeled two human balancing tasks: stick balancing and quiet stance. The balancing activity of humans was considered by a proportional-integral-derivative-acceleration delayed feedback controller for both models. Stabilizability diagrams were determined where the loss of stabilizability is associated with the loss of balance of humans. I carried out comparison between the calculated and the experimental results.

Thesis 7

By modeling the human balancing process with a proportional-integral-derivative-acceleration (PIDA) delayed feedback controller in stick balancing and quiet stance, stabilizability diagrams can be computed. These diagrams present the domain of system and control parameters, for which the unstable equilibrium of the open-loop system becomes stable in the closed-loop system. The comparison of the results to those of the literature shows, that in case of the quiet stance model, there always exists a stabilizing set of control parameters. In contrast, for the stick balancing model, there always exists a critical stick length below which the stick cannot be stabilized for any set of control parameters. The integral gain of the control loop cannot improve the stabilizability properties of the investigated models of the balancing tasks.

Related publications: [45]

Appendix A

Results from numerical analysis

A.1 Legendre polynomials

Definition of Legendre polynomials by Bonnet's recursion formula:

$$\begin{aligned} P_0(\zeta) &= 1 \\ P_1(\zeta) &= \zeta \\ P_j(\zeta) &= \frac{2j-1}{j} \zeta P_{j-1}(\zeta) - \frac{j-1}{j} P_{j-2}(\zeta) \quad j = 2, 3, \dots \end{aligned}$$

Some properties of Legendre polynomials:

- orthogonality: $\int_{-1}^1 P_j(\zeta) P_i(\zeta) d\zeta = \frac{2}{2j+1} \delta_{i,j}$
- $P_j(1) = 1$ and $P_j(-1) = (-1)^j$

Recursion formula for the first derivative of Legendre polynomials:

$$P'_j(\zeta) = (2j-1) P_{j-1}(\zeta) + P'_{j-2}(\zeta) \quad j = 2, 3, \dots$$

Due to the above formula and the orthogonality property of Legendre polynomials:

$$\int_{-1}^1 P'_j(\zeta) P_i(\zeta) d\zeta = \begin{cases} 0 & \text{if } i > j \\ 2 & \text{if } (j-i) \bmod 2 \neq 0 \\ 0 & \text{if } (j-i) \bmod 2 = 0 \end{cases}$$

A.2 Lobatto-type Legendre–Gauss quadrature

The Lobatto-type Legendre–Gauss quadrature approximates a definite integral by a sum as

$$I = \int_a^b x(t) dt \approx \tilde{I} = \sum_{q=1}^{n+1} x(t_q) w_q,$$

where $t_q = \frac{a-b}{2}\zeta_q^* + \frac{a+b}{2}$ with ζ_q^* and w_q being the quadrature nodes and weights, respectively. The Lobatto-type Legendre–Gauss quadrature gives exact results for all polynomials with maximum order $2n - 1$. The quadrature nodes are the roots of $(1 - \zeta^2)P'_n(\zeta)$, that is $-1, 1$ and the roots of the first derivative of the Legendre polynomial of order n . The quadrature weights are given by

$$w_q = \begin{cases} \frac{2}{n(n+1)} & \text{if } q = 1, n+1; \\ \frac{2}{n(n+1)P_n^2(\zeta_q^*)} & \text{if } q = 2, 3, \dots, n. \end{cases}$$

A.3 Chebyshev points and polynomials

Theorem

Consider the Lagrange interpolant $\tilde{x}(\theta)$ of $x(\theta)$ on the domain $\theta \in [a, b]$, using the node set $\{\theta_j\}_{j=1}^n \subset [a, b]$. Assume, that $x(t)$ is C^n on the given interval. Then at any θ point of $[a, b]$, the error $E_n(\theta) = x(\theta) - \tilde{x}(\theta)$ of interpolation is given by the formula

$$E_n(\theta) = -\frac{x^{(n)}(\theta^*)}{n!} \omega_n(\theta), \quad \omega_n(\theta) = \prod_{j=1}^n (\theta - \theta_j),$$

where $\theta^* \in [a, b]$ is a constant which depends on the value of θ .

Theorem

Let $[a, b]$ a fixed interval. For a $p_n(\theta)$ n -order polynomial with leading coefficient 1, $|p_n(\theta)|_{C[a,b]} \leq |q_n(\theta)|_{C[a,b]}$ for all $q_n(\theta)$ n -order polynomial with leading coefficient 1 if and only if $p_n(\theta)$ has at least $n + 1$ distinct absolute extrema on $[a, b]$. On these points the absolute value of $p_n(\theta)$ is the same, while the sign of its value alternates.

Theorem

There is only one $p_n(\theta)$ n -order polynomial with leading coefficient 1 on $[a, b]$, for which $|p_n(\theta)|_{C[a,b]} \leq |q_n(\theta)|_{C[a,b]}$ for all $q_n(\theta)$ n -order polynomial with leading coefficient 1.

Theorem

On the domain $[-1, 1]$ the function

$$\tilde{T}_{n+1}(\theta) = \frac{\cos(n \arccos(\theta))}{2^{n-1}}, \quad n \geq 1$$

is an n -order polynomial with leading coefficient 1 having $n + 1$ absolute extrema with alternating signs.

Definition

Polynomials $T_1(\theta) = 1$ and $T_{n+1}(\theta) = 2^{n-1} \tilde{T}_{n+1}$, $n \geq 1$ are called Chebyshev polynomials.

Corollary

Since $\omega_n(\theta)$ is an n -order polynomial with leading coefficient 1, choosing $\{\theta_j\}_{j=1}^n$ as the set of zeros of $T_{n+1}(\theta)$ scaled from domain $[-1, 1]$ to $[a, b]$ leads to minimal $|E_n(\theta)|_{C[a,b]}$, moreover an upper estimation can be given by

$$|E_n(\theta)|_{C[a,b]} \leq \frac{\max_{\theta^* \in [a,b]} \{|x^{(n)}(\theta^*)|\}}{n! 2^{n-1}}.$$

Theorem

The Chebyshev polynomial $T_{n+1}(\theta) = \cos(n \arccos(\theta))$ can be given by the recursive formula

$$T_{j+1}(\theta) = 2T_j(\theta) - T_{j-1}(\theta), \quad j = 2, \dots, n;$$

with $T_1(\theta) = 1$ and $T_2(\theta) = \theta$.

Theorem

If $x(\theta)$ is absolute continuous on the domain of interpolation, then its Lagrange interpolant $\tilde{x}(\theta)$ on the zeros of the Chebyshev nodes converges uniformly to $x(\theta)$ in the $C[a, b]$ norm as $n \rightarrow \infty$.

Bibliography

- [1] M. Abramowitz and I. A. Stegun. *Handbook of mathematical functions with formulas, graphs and mathematical tables*. 10th ed. Washington D.C.: National Bureau of Standards, 1972.
- [2] Y. Altintas. *Manufacturing Automation*. New York: Cambridge University Press, 2012.
- [3] Y. Altintas and E. Budak. "Analytical prediction of stability lobes in milling". In: *CIRP Annals Manufacturing Technology* 44 (1995), pp. 357–362.
- [4] Y. Asai, S. Tateyama, and T. Nomura. "Learning an intermittent control strategy for postural balancing using an EMG-based human-computer interface". In: *Plos One* 8(5).e62956 (2013), pp. 1–19.
- [5] D. Bachrathy and G. Stepan. "Bisection method in higher dimensions and the efficiency number". In: *Periodica Polytechnica Mechanical Engineering* 56(2) (2012), pp. 81–86.
- [6] D. Bachrathy and G. Stepan. "Improved prediction of stability lobes with extended multi frequency solution". In: *CIRP Annals Manufacturing Technology* 62 (2013), pp. 411–414.
- [7] J. P. Berrut and F. L. Trefethen. "Barycentric Lagrange Interpolation". In: *SIAM Review* 46(3) (2004), pp. 501–517.
- [8] O. A. Bobrenkov et al. "Analysis of milling dynamics for simultaneously engaged cutting teeth". In: *Journal of Sound and Vibration* 329(5) (2010), pp. 585–606.
- [9] D. Breda, S. Maset, and R. Vermiglio. "An adaptive algorithm for efficient computation of level curves of surfaces". In: *Numerical Algorithms* 52 (2009), pp. 605–628.
- [10] D. Breda, S. Maset, and R. Vermiglio. "Approximation of eigenvalues of evolution operators for linear retarded functional differential equations". In: *SIAM Journal on Numerical Analysis* 50(3) (2009), pp. 1456–1483.
- [11] D. Breda, S. Maset, and R. Vermiglio. "Pseudospectral differencing methods for characteristic roots of delay differential equations". In: *SIAM Journal on Scientific Computing* 27(2) (2005), pp. 482–495.
- [12] D. Breda, S. Maset, and R. Vermiglio. *Stability of Linear Delay Differential Equations*. New York: Springer, 2015.
- [13] E. A. Butcher and O. A. Bobrenkov. "On the Chebyshev spectral continuous time approximation for constant and periodic delay differential equations". In: *Communications in Nonlinear Science and Numerical Simulation* 16 (2011), pp. 1541–1554.
- [14] J. L. Cabrera and J. Milton. "Stick balancing: On-off intermittency and survival times". In: *Nonlinear Studies* 11(3) (2004), pp. 305–317.
- [15] J. L. Cabrera and J. G. Milton. "Stick balancing, falls and Dragon-Kings". In: *The European Physical Journal Special Topics* 205(1) (2012), pp. 231–241.

- [16] O. Diekmann et al. *Delay Equations: Functional-, Complex-, and Nonlinear Analysis*. New York: Springer-Verlag, 1995.
- [17] N. J. M. van Dijk et al. "Robust active chatter control in the high-speed milling process". In: *IEEE Transactions on Control Systems Technology* 20(4) (2012), pp. 901–917.
- [18] Y. Ding et al. "A full-discretization method for prediction of milling stability". In: *International Journal of Machine Tools and Manufacture* 50 (2010), pp. 502–509.
- [19] Y. Ding et al. "Numerical Integration Method for Prediction of Milling Stability". In: *Journal of Manufacturing Science and Engineering* 133.031005 (2011), pp. 1–9.
- [20] Y. Ding et al. "Second-order full discretization method for milling stability prediction". In: *International Journal of Machine Tools and Manufacture* 50 (2010), pp. 926–932.
- [21] Y. Ding et al. "Stability Analysis of Milling Via the Differential Quadrature Method". In: *Journal of Manufacturing Science and Engineering* 135.044502 (2013), pp. 1–7.
- [22] Z. Dombovari et al. "Prediction of multiple dominant chatter frequencies in milling processes". In: *International Journal of Machine Tools and Manufacture* 51 (2011), pp. 457–464.
- [23] D. Gottlieb and S. A. Orszag. *Numerical analysis of spectral methods*. Philadelphia: SIAM, 1977.
- [24] G. Habib, G. Rega, and G. Stepan. "Delayed digital position control of a single-DoF system and the nonlinear behavior of the act-and-wait controller". In: *Journal of Vibration and Control* 22(2) (2016), pp. 481–495.
- [25] J. K. Hale and S. M. V. Lunel. *Introduction to functional differential equations*. New York: Springer-Verlag, 1993.
- [26] T. Insperger, D. Lehotzky, and G. Stepan. "Regenerative delay, parametric forcing and machine tool chatter: A review". In: *IFAC-PapersOnLine* 48(12) (2015), pp. 322–327.
- [27] T. Insperger and J. Milton. "Sensory uncertainty and stick balancing at the fingertip". In: *Biological Cybernetics* 108(1) (2014), pp. 85–101.
- [28] T. Insperger, J. Milton, and G. Stepan. "Acceleration feedback improves balancing against reflex delay". In: *Journal of the Royal Society Interface* 10.20120763 (2013).
- [29] T. Insperger and G. Stepan. *Semi-discretization for time-delay systems*. New York: Springer, 2011.
- [30] T. Insperger and G. Stepan. "Semi-discretization method for delayed systems". In: *International Journal for Numerical Methods in Engineering* 55(5) (2002), pp. 503–518.
- [31] T. Insperger and G. Stepan. "Stability chart for the delayed Mathieu equation". In: *Proceedings of the Royal Society London A* 458 (2002), pp. 1989–1998.
- [32] T. Insperger and G. Stepan. "Updated semi-discretization method for periodic delay-differential equations with discrete delay". In: *International Journal for Numerical Methods in Engineering* 61 (2004), pp. 117–141.

- [33] K. Ito and R. Teglas. "Legendre-tau approximations for functional differential equations". In: *SIAM Journal of Control and Optimization* 24(4) (1986), pp. 737–759.
- [34] M. I. Jordan. "Computational aspects of motor control and motor learning". In: *Handbook of perception and action: motor skills*. Ed. by H. Heuer and S. Keele. New York: Academic Press, 1996, pp. 1–64.
- [35] M. Kawato, K. Furukawa, and R. Suzuki. "A Hierarchical neural-network model for control and learning of voluntary movement". In: *Biological Cybernetics* 57(3) (1987), pp. 169–185.
- [36] F. A. Khasawneh and B. P. Mann. "A spectral element approach for the stability analysis of time-periodic delay equations with multiple delays". In: *Communications in Nonlinear Science and Numerical Simulation* 18 (2013), pp. 2129–2141.
- [37] F. A. Khasawneh and B. P. Mann. "A spectral element approach for the stability of delay systems". In: *International Journal for Numerical Methods in Engineering* 87 (2011), pp. 566–592.
- [38] F. A. Khasawneh and B. P. Mann. "Stability of delay integro-differential equations using a spectral element method". In: *Mathematical and Computer Modelling* 54 (2011), pp. 2493–2503.
- [39] O. Kienzle. "Spezifische schnittkräfte bei der metallbearbeitung". In: *Werkstattstechnik und Maschinenbau* 47 (1957), pp. 224–225.
- [40] L. L. Kovacs and J. Kovecses. "Dynamics of coupled haptic systems". In: *Proceedings of the IEEE World Haptics Conference, Evanston, USA*. 2015, pp. 286–292.
- [41] C. Lanczos. *Applied Analysis*. New Jersey: Prentice Hall, 1956.
- [42] K. Y. Lee et al. "A new paradigm for human stick balancing: a suspended not an inverted pendulum". In: *Experimental Brain Research* 221 (2012), pp. 309–328.
- [43] D. Lehoczky and T. Insperger. "A least-square spectral element method for stability analysis of time delay systems". In: *IFAC-PapersOnLine* 48(12) (2015), pp. 382–385.
- [44] D. Lehoczky and T. Insperger. "A pseudospectral tau approximation for time delay systems and its comparison with other weighted-residual-type methods". In: *International Journal for Numerical Methods in Engineering* 108 (2016), pp. 588–613.
- [45] D. Lehoczky and T. Insperger. "Az emberi egyensúlyozás mechanikai modelllezése PIDA szabályozó segítségével". In: *Biomechanica Hungarica* 7(1) (2014), pp. 24–33.
- [46] D. Lehoczky and T. Insperger. "Stability of delayed oscillators subjected to digital PD control". In: *IFAC Proceedings Volumes* 45(14) (2012), pp. 73–78.
- [47] D. Lehoczky and T. Insperger. "Stability of turning processes subjected to digital PD control". In: *Periodica Polytechnica Mechanical Engineering* 56(1) (2012), pp. 33–42.
- [48] D. Lehoczky, T. Insperger, and G. Stepan. "Extension of the spectral element method for stability analysis of time-periodic delay-differential equations with multiple and distributed delays". In: *Communications in Nonlinear Science and Numerical Simulation* 35 (2016), pp. 177–189.

- [49] D. Lehotzky, T. Insperger, and G. Stepan. "Stability of time-periodic hybrid time-delay systems with applications to controlled milling operations". In: *Nonlinear Analysis: Hybrid Systems* (2016). submitted.
- [50] D. Lehotzky, T. Insperger, and G. Stepan. "State-dependent, non-smooth model of chatter vibrations in turning". In: *Proceedings of the ASME IDETC/CIE Conference, Boston, USA*. DETC2015-46748. 2015, pp. 1–8.
- [51] D. Lehotzky, T. Insperger, and G. Stépán. "Szán szabályozásának hatása az esztergálás regeneratív rezgéseire". In: *XII. Magyar Mechanikai Konferencia CD Kiadványa*. 234. 2015, pp. 1–8.
- [52] D. Lehotzky and J. Turi. "Increasing the stability limits of turning processes by using digital control". In: *Mathematics in Engineering, Science and Aerospace* 4(2) (2013), pp. 97–115.
- [53] D. Lehotzky, J. Turi, and T. Insperger. "Stabilizability diagram for turning processes subjected to digital PD control". In: *International Journal of Dynamics and Control* 2 (2014), pp. 46–54.
- [54] D. Lehotzky et al. "Spectral element method for stability analysis of milling processes with discontinuous time-periodicity". In: *International Journal of Advanced Manufacturing Technology* (2016). published online. DOI: [10.1007/s00170-016-9044-z](https://doi.org/10.1007/s00170-016-9044-z).
- [55] Y. Liu, D. Zhang, and B. Wu. "An efficient full-discretization method for prediction of milling stability". In: *International Journal of Machine Tools and Manufacture* 63 (2012), pp. 44–48.
- [56] B. P. Mann et al. "Simultaneous Stability and Surface Location Error Predictions in Milling". In: *Journal of Manufacturing Science and Engineering* 127 (2005), pp. 446–453.
- [57] C. Maurer and R. J. Peterka. "A new interpretation of spontaneous sway measures based on a simple model of human postural control". In: *Journal of Neurophysiology* 93(1) (2005), pp. 189–200.
- [58] B. Mehta and S. Schaal. "Forward models in visuomotor control". In: *Journal of Neurophysiology* 88(2) (2002), pp. 942–953.
- [59] D. Mei et al. "Magneto-rheological fluid-controlled boring bar for chatter suppression". In: *Journal of Materials Processing Technology* 209 (2009), pp. 1861–1870.
- [60] W. Michiels and S. I. Niculescu. *Stability and Stabilization of Time-Delay Systems*. Philadelphia: SIAM, 2007.
- [61] W. Michiels et al. "Continuous pole placement for delay equations". In: *Automatica* 38 (2002), pp. 747–761.
- [62] J. Milton et al. "The time-delayed inverted pendulum: Implications for human balance control". In: *Chaos* 19.026110 (2009), pp. 1–12.
- [63] F. Moss and J. G. Milton. "Medical technology - Balancing the unbalanced". In: *Nature* 425(6961) (2003), pp. 911–912.
- [64] J. Munoa et al. "Active suppression of structural chatter vibrations using machine drives and accelerometers". In: *CIRP Annals - Manufacturing Technology* 64 (2015), pp. 385–388.
- [65] J. Munoa et al. "Chatter suppression in ram type travelling column milling machines using a biaxial inertial actuator". In: *CIRP Annals - Manufacturing Technology* 62 (2013), pp. 407–410.

- [66] J. B. Niu et al. "Runge-Kutta methods for a semi-analytical prediction of milling stability". In: *Nonlinear Dynamics* 76 (2014), pp. 289–304.
- [67] G. Orosz, R. E. Wilson, and G. Stepan. "Traffic jams: dynamics and control". In: *Philosophical Transactions of the Royal Society A* 368 (2010), pp. 4455–4479.
- [68] G. Orosz et al. "Exciting traffic jams: nonlinear phenomena behind traffic jam formation on highways". In: *Physical Review E* 80(4).046205 (2009), pp. 1–12.
- [69] G. Quintana and J. Ciurana. "Chatter in machining processes: A review". In: *International Journal of Machine Tools and Manufacture* 51 (2011), pp. 363–376.
- [70] Q. Quo, Y. Sun, and Y. Jiang. "On the accurate calculation of milling stability limits using third-order full-discretization method". In: *International Journal of Machine Tools and Manufacture* 62 (2012), pp. 61–66.
- [71] K. J. Åström and R. M. Murray. *Feedback Systems*. New Jersey: Princeton University Press, 2008.
- [72] Fujita H Shiraiishi M Yamanaka K. "Optimal control of chatter in turning". In: *International Journal of Machine Tools and Manufacture* 31(1) (1991), pp. 31–43.
- [73] N. D. Sims. "Vibration absorbers for chatter suppression: A new analytical tuning methodology". In: *Journal of Sound and Vibration* 301 (2007), pp. 592–607.
- [74] G. Stepan. "Modelling nonlinear regenerative effects in metal cutting". In: *Philosophical Transactions of the Royal society A* 359 (2001), pp. 739–757.
- [75] G. Stepan. *Retarded dynamical systems*. Horlow: Longman, 1989.
- [76] G. Stepan. "Vibrations of machines subjected to digital force control". In: *International Journal of Solids and Structures* 38 (2001), pp. 2149–2159.
- [77] G. Stepan, Z. Dombovari, and J. Munoa. "Identification of cutting force characteristics based on chatter experiments". In: *CIRP Annals Manufacturing Technology* 60 (2011), pp. 113–116.
- [78] G. Stepan et al. "Cylindrical milling tools: Comparative real case study for process stability". In: *CIRP Annals Manufacturing Technology* 63 (2014), pp. 385–388.
- [79] J. Q. Sun. "A method of continuous time approximation of delayed dynamical systems". In: *Communications in Nonlinear Science and Numerical Simulation* 14 (2009), pp. 998–1007.
- [80] Y. Suzuki et al. "Intermittent control with ankle, hip, and mixed strategies during quiet standing: A theoretical proposal based on a double inverted pendulum model". In: *Journal of Theoretical Biology* 310 (2012), pp. 55–79.
- [81] D. Takacs and G. Stepan. "Contact patch memory of tyres leading to lateral vibrations of four-wheeled vehicles". In: *Philosophical Transactions of the Royal society A* 371.20120427 (2013), pp. 1–12.
- [82] Y. S. Tarng, J. Y. Kao, and E. C. Lee. "Chatter suppression in turning operations with a tuned vibration absorber". In: *Journal of Materials Processing Technology* 105 (2000), pp. 55–60.
- [83] J. Tlusty et al. *Selbsterregte Schwingungen an Werkzeugmaschinen*. Berlin: VEB Verlag Technik, 1962.
- [84] S. A. Tobias. *Machine Tool Vibration*. London: Blackie, 1965.

- [85] L. N. Trefethen. "Is Gauss quadrature better than Clenshaw–Curtis?" In: *SIAM Review* 50(1) (2008), pp. 67–87.
- [86] C. P. Vyasarayani, S. Subhash, and T. Kalmar-Nagy. "Spectral approximations for characteristic roots of delay differential equations". In: *International Journal of Dynamics and Control* 2 (2014), pp. 126–132.
- [87] P. Wahi and A. Chatterjee. "Galerkin projections for delay differential equations". In: *Journal of Dynamic Systems, Measurement, and Control* 127 (2005), pp. 80–87.
- [88] P. Wahi and A. Chatterjee. "Self-interrupted regenerative metal cutting in turning". In: *International Journal of Non-Linear Mechanics* 43 (2008), pp. 111–123.
- [89] D. A. Winter et al. "Stiffness control of balance in quiet standing". In: *Journal of Neurophysiology* 80(3) (1998), pp. 1211–1221.
- [90] M. H. Woollacott, C. Vonhosten, and B. Rosblad. "Relation between muscle response onset and body segmental movements during postural perturbations in humans". In: *Experimental Brain Research* 72(2) (1988), pp. 593–604.
- [91] Y. Yang, J. Munoa, and Y. Altintas. "Optimization of multiple tuned mass dampers to suppress machine tool chatter". In: *International Journal of Machine Tools and Manufacture* 50 (2010), pp. 834–842.
- [92] M. Zatarain and Z. Dombovari. "Stability analysis of milling with irregular pitch tools by the implicit subspace iteration method". In: *International Journal of Dynamics and Control* 2 (2014), pp. 26–34.
- [93] M. Zatarain et al. "Stability of milling processes with continuous spindle speed variation: Analysis in the frequency and time domains, and experimental correlation". In: *CIRP Annals-Manufacturing Technology* 57 (2008), pp. 379–384.
- [94] M. X. Zhao and B. Balachandran. "Dynamics and stability of milling process". In: *International Journal of Solids and Structures* 38 (2001), pp. 2233–2248.