Acta Mech Sin (2015) 31(3):406–415
DOI 10.1007/s10409-015-0397-6

RESEARCH PAPER

A high-order full-discretization method using Hermite interpolation for periodic time-delayed differential equations

Yilong Liu[1] · **Achim Fischer**[2] · **Peter Eberhard**[2] · **Baohai Wu**[1]

Received: 14 May 2014 / Revised: 20 October 2014 / Accepted: 16 February 2015 / Published online: 29 May 2015

Abstract A high-order full-discretization method (FDM) using Hermite interpolation (HFDM) is proposed and implemented for periodic systems with time delay. Both Lagrange interpolation and Hermite interpolation are used to approximate state values and delayed state values in each discretization step. The transition matrix over a single period is determined and used for stability analysis. The proposed method increases the approximation order of the semidiscretization method and the FDM without increasing the computational time. The convergence, precision, and efficiency of the proposed method are investigated using several Mathieu equations and a complex turning model as examples. Comparison shows that the proposed HFDM converges faster and uses less computational time than existing methods.

Keywords Full-discretization method · Time delay · Stability · Chatter

1 Introduction

Time-periodic delay differential equations (PDDEs) are widely reported in the literature. Their stability properties are important in many recent studies. Many applications of PDDEs are found in mathematics and engineering. For example, Mathieu equations [1], which combine the effect of parametric excitation on a delayed and damped oscillator, are a special second-order linear PDDEs. Their stability and resonance behavior were investigated in many recent studies [2–4]. In engineering, one of the most important applications is the cutting dynamics in connection with regenerative chatter [5–8]. Other cases of PDDEs can also be found in remote force reflective manipulations or human–machine systems [9–12], to name just a few.

Three categories of methods exist for analyzing the stability of these equations: frequency-dependent methods, semidiscretization methods (SDMs) or full-discretization methods (FDMs), and time domain simulations. In this paper, we focus on making further improvements to the discretization methods, which are well suited for analyzing periodic systems with delays while maintaining a reasonable computational time. In this category, the semidiscretization method (SDM) was first introduced by Insperger and Stépán [13]. With the SDM, only the delayed terms are discretized, while the actual time domain terms are unchanged, in contrast to the FDM. An updated SDM was subsequently proposed [14], where the delayed term is approximated as a weighted sum of two neighboring discrete delayed state values. The updated SDM can be used efficiently, for example, to construct stability charts of Mathieu equations or 1-degree-of-freedom (DOF) and 2-DOF turning or milling processes.

To further improve the efficiency of the method, a higher-order SDM was proposed [15]. Using several Mathieu equations as examples, it was proven that a first-order SDM performs better than the updated SDM. A FDM [16] was developed that was shown to converge faster. This method can also be seen as a modified SDM. It was applied to

✉ Yilong Liu
achim.fischer@itm.uni-stuttgart.de

[1] Key Laboratory of Contemporary Design and Integrated Manufacturing Technology, Northwestern Polytechnical University, 710072 Xi'an, China

[2] Institute of Engineering and Computational Mechanics, University of Stuttgart, Pfaffenwaldring 9, 70569 Stuttgart, Germany

analyze milling processes. Later on, other methods were introduced [17]. However, these methods are focused only on solving DDEs related to the milling process. The original SDM is a classic method. Though not the most efficient in some cases, it is a relatively robust and widely used algorithm.

The construction of stability charts using the existing methods still requires considerable CPU time. Thus new developments in the field of stability analysis of delay systems are still important. In this paper, we propose an efficient FDM based on higher-order approximations using Hermite and Lagrangian interpolation (HFDM). This HFDM is subsequently used to investigate the damped and delayed Mathieu equation. Three different cases, where the time delay is larger than, the same as, and smaller than the time period, are investigated. Furthermore, a more complex model of the turning of thin-walled cylinders is taken into consideration [18,19]. The higher system dimension and comparatively long delays of this problem pose a challenging task. To compare the computational efficiency and precision, several other existing methods are compared with the proposed method.

2 Stability analysis using the full-discretization method

Stability analysis of a system of delay differential equations (DDEs) has attracted and still attracts considerable interest in the scientific community. In this work, we consider time-periodic DDEs of the form

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_0\boldsymbol{x}(t) + \boldsymbol{A}_1(t)\boldsymbol{x}(t) - \boldsymbol{B}(t)\boldsymbol{x}(t-\tau). \tag{1}$$

The matrix $\boldsymbol{A}(t)$ of the system is hereby separated into a constant term $\boldsymbol{A}_0$ and a time-dependent part $\boldsymbol{A}_1(t)$. The input matrix $\boldsymbol{B}(t)$ projects the influence of the system past on the state vector $\boldsymbol{x}$ at the current timepoint. Both $\boldsymbol{A}_1(t)$ and $\boldsymbol{B}(t)$ are time periodic with periodicity T, i.e,

$$\boldsymbol{A}_1(t+T) = \boldsymbol{A}_1(t), \qquad \boldsymbol{B}(t+T) = \boldsymbol{B}(t). \tag{2}$$

Although Floquet theory [20] can be applied to systems with time delays, practical application is not straightforward. The introduction of the time delay in a time-periodic dynamical system causes the phase space to grow from finite-dimensional to infinite-dimensional, which leads to an infinite-dimensional monodromy matrix.

To solve the problem, a discrete map of the infinite-dimensional, continuous space must be found. This is done through an approximating discrete system whose state vector $\boldsymbol{\Xi}$ contains the current and a finite number of past states of the system. What is determined is, thus, the transition between two successive periods of the approximated discrete system,

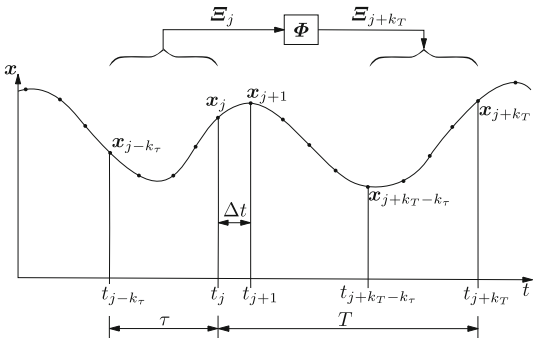

Fig. 1 Approximation as a time-discrete system

$$\boldsymbol{\Xi}_{j+k_T} = \boldsymbol{\Phi}\boldsymbol{\Xi}_j, \tag{3}$$

with

$$\boldsymbol{\Xi}_j = \begin{bmatrix} \boldsymbol{x}_j \\ \boldsymbol{x}_{j-1} \\ \cdots \\ \boldsymbol{x}_{j-k_\tau} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\Xi}_{j+k_T} = \begin{bmatrix} \boldsymbol{x}_{j+k_T} \\ \boldsymbol{x}_{j+k_T-1} \\ \vdots \\ \boldsymbol{x}_{j+k_T-k_\tau} \end{bmatrix}, \tag{4}$$

as shown in Fig. 1.

The stability of the approximated system can be determined by means of the eigenvalues of the transition matrix $\boldsymbol{\Phi}$. For a stable system, they must be located inside the unit circle of the complex plane:

$$|\lambda(\boldsymbol{\Phi})|_{\max} = \begin{cases} > 1, & \text{unstable,} \\ = 1, & \text{boundary of stability,} \\ < 1, & \text{stable.} \end{cases} \tag{5}$$

A number of different discretization techniques exists that seek to find a suitable discrete approximation of a continuous time-varying system and the corresponding transition matrix. They all follow a similar approach. First, an approximated solution is found on each of the subintervals. These local solutions are then combined to obtain a solution for the transition (3) between two successive states of the discrete system. The FDM proposed in this work uses a finite number of past system states in conjunction with Hermite and Lagrangian interpolation to approximate the infinite-dimensional state of the system with delay. In what follows, the system is discretized into k_T time steps of length Δt. The period resolution k_T is chosen such that $T = k_T \Delta t$.

The time delay τ equals the period of the system, and it is $T = \tau$ in the case of turning. In the milling case, T is usually a multiple of τ, i.e., $T = n_t\tau$, where n_t is the number of teeth. For more general DDEs, the time period T and the time delay τ are not always equal or integer multiples. Therefore, we introduce another approximation parameter, k_τ, related to the time delay. The relationships between k_T and k_τ are

$$k_T = \begin{cases} \text{int}\left(k_\tau \frac{T}{\tau}\right), & \text{for } T > \tau, \\ k_\tau, & \text{for } T = \tau, \end{cases}$$
$$k_\tau = \text{int}\left(k_T \frac{\tau}{T}\right), \quad \text{for } T < \tau. \tag{6}$$

Subsequently, an approximation on the interval $[t_j, t_{j+1}]$ is derived and the transition from $\boldsymbol{x}_j$ to $\boldsymbol{x}_{j+1}$ determined.

The time-dependent matrices $\boldsymbol{A}_1(t)$ and $\boldsymbol{B}(t)$ are approximated by a constant value on each interval. We found that the approximation of the time-varying matrices had only a slight effect on the overall quality of the results compared to the approximation of the state term. The matrices $\boldsymbol{A}_1(t)$ and $\boldsymbol{B}(t)$ are therefore approximated by their value at the beginning of the corresponding interval:

$$\boldsymbol{A}_1(t) \approx \boldsymbol{A}_1\left(t_j\right) := \boldsymbol{A}_{1,j} \quad \text{and} \quad \boldsymbol{B}(t) \approx \boldsymbol{B}\left(t_j\right) := \boldsymbol{B}_j. \tag{7}$$

The evolution of the system state $\boldsymbol{x}(t)$ on the interval $[t_j, t_{j+1}]$ is approximated using a Hermite polynomial of the form

$$\begin{aligned} \boldsymbol{x}(t) \approx \tilde{\boldsymbol{x}}(t) = {} & h_{00}(t)\boldsymbol{x}(t_j) + h_{10}(t)\boldsymbol{x}(t_{j+1}) \\ & + h_{01}(t)\dot{\boldsymbol{x}}(t_j) + h_{11}(t)\dot{\boldsymbol{x}}(t_{j+1}), \end{aligned} \tag{8}$$

with the coefficients

$$\begin{aligned} h_{00}(t) &= \left(1 + 2\frac{t - t_j}{t_{j+1} - t_j}\right)\left(\frac{t - t_{j+1}}{t_j - t_{j+1}}\right)^2, \\ h_{10}(t) &= \left(1 + 2\frac{t - t_{j+1}}{t_j - t_{j+1}}\right)\left(\frac{t - t_j}{t_{j+1} - t_j}\right)^2, \\ h_{01}(t) &= \left(t - t_j\right)\left(\frac{t - t_{j+1}}{t_j - t_{j+1}}\right)^2, \\ h_{11}(t) &= \left(t - t_{j+1}\right)\left(\frac{t - t_j}{t_{j+1} - t_j}\right)^2. \end{aligned} \tag{9}$$

The derivatives $\dot{\boldsymbol{x}}(t_j)$ and $\dot{\boldsymbol{x}}(t_{j+1})$ in Eq. (8) are obtained from the state equation (1):

$$\begin{aligned} \dot{\boldsymbol{x}}(t_j) &= \boldsymbol{A}_0\boldsymbol{x}(t_j) + \boldsymbol{A}_{1,j}\boldsymbol{x}(t_j) - \boldsymbol{B}_j\boldsymbol{x}(t_j - \tau), \\ \dot{\boldsymbol{x}}(t_{j+1}) &= \boldsymbol{A}_0\boldsymbol{x}(t_{j+1}) + \boldsymbol{A}_1(t_{j+1})\boldsymbol{x}(t_{j+1}) \\ &\quad - \boldsymbol{B}(t_{j+1})\boldsymbol{x}(t_{j+1} - \tau), \\ &\approx \boldsymbol{A}_0\boldsymbol{x}(t_{j+1}) + \boldsymbol{A}_{1,j}\boldsymbol{x}(t_{j+1}) - \boldsymbol{B}_j\boldsymbol{x}(t_{j+1} - \tau). \end{aligned} \tag{10}$$

The evolution of past states $\boldsymbol{x}(t-\tau)$ is approximated using a quadratic interpolation and the discrete values $\boldsymbol{x}(t_j - \tau)$, $\boldsymbol{x}(t_{j+1} - \tau)$, and $\boldsymbol{x}(t_{j+2} - \tau)$:

$$\begin{aligned} \boldsymbol{x}(t - \tau) \approx \tilde{\boldsymbol{x}}_\tau(t) = {} & l_1(t)\boldsymbol{x}(t_j - \tau) + l_2(t)\boldsymbol{x}(t_{j+1} - \tau) \\ & + l_3(t)\boldsymbol{x}(t_{j+2} - \tau), \end{aligned} \tag{11}$$

where the coefficients of the polynomial are given by

$$\begin{aligned} l_1(t) &= \frac{t - t_{j+1}}{t_j - t_{j+1}}\frac{t - t_{j+2}}{t_j - t_{j+2}}, \\ l_2(t) &= \frac{t - t_j}{t_{j+1} - t_j}\frac{t - t_{j+2}}{t_{j+1} - t_{j+2}}, \\ l_3(t) &= \frac{t - t_j}{t_{j+2} - t_j}\frac{t - t_{j+1}}{t_{j+2} - t_{j+1}}. \end{aligned} \tag{12}$$

Using Eqs. (7), (8) and (11), an approximating ordinary differential equation (ODE) of the time-varying delay differential equation (1) can be found:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_0\boldsymbol{x}(t) + \boldsymbol{A}_{1,j}\tilde{\boldsymbol{x}}(t) - \boldsymbol{B}_j\tilde{\boldsymbol{x}}_\tau(t), \quad t \in [t_j, t_{j+1}]. \tag{13}$$

Using this equation in a time integration gives the state $\boldsymbol{x}_{j+1}$ at the end of the subinterval. The solution takes the form

$$\begin{aligned} \boldsymbol{Q}_j\boldsymbol{x}_{j+1} = {} & \boldsymbol{H}_j\boldsymbol{x}_j + \boldsymbol{H}_{j-k_\tau}\boldsymbol{x}_{j-k_\tau} \\ & + \boldsymbol{H}_{j+1-k_\tau}\boldsymbol{x}_{j+1-k_\tau} + \boldsymbol{H}_{j+2-k_\tau}\boldsymbol{x}_{j+2-k_\tau}, \end{aligned} \tag{14}$$

where

$$\begin{aligned} \boldsymbol{Q}_j = {} & \boldsymbol{I} - \left(-\frac{\boldsymbol{\Psi}_2}{\Delta t} + \frac{\boldsymbol{\Psi}_3}{\Delta t^2}\right)\boldsymbol{A}_{1,j}\boldsymbol{A}_{1,j} \\ & - \left(\frac{\boldsymbol{\Psi}_3\boldsymbol{A}_0}{\Delta t^2} + 3\frac{\boldsymbol{\Psi}_2}{\Delta t^2} - \frac{\boldsymbol{\Psi}_2\boldsymbol{A}_0}{\Delta t} - 2\frac{\boldsymbol{\Psi}_3}{\Delta t^3}\right)\boldsymbol{A}_{1,j}, \end{aligned} \tag{15}$$

$$\boldsymbol{H}_j = + \left(\boldsymbol{\Psi}_1 + \frac{\boldsymbol{\Psi}_3}{\Delta t^2} - 2\frac{\boldsymbol{\Psi}_2}{\Delta t}\right)\boldsymbol{A}_{1,j}\boldsymbol{A}_{1,j} + \mathrm{e}^{\boldsymbol{A}_0\Delta t} \tag{16}$$

$$\begin{aligned} & + \left(\boldsymbol{\Psi}_1\boldsymbol{A}_0 + 2\frac{\boldsymbol{\Psi}_3}{\Delta t^3} - 2\frac{\boldsymbol{\Psi}_2\boldsymbol{A}_0}{\Delta t}\right. \\ & \left. + \frac{\boldsymbol{\Psi}_3\boldsymbol{A}_0}{\Delta t^2} - 3\frac{\boldsymbol{\Psi}_2}{\Delta t^2} + \boldsymbol{\Psi}_0\right)\boldsymbol{A}_{1,j}, \end{aligned} \tag{17}$$

$$\boldsymbol{H}_{j-k_\tau} = \left[\left(2\frac{\boldsymbol{\Psi}_2}{\Delta t} - \frac{\boldsymbol{\Psi}_3}{\Delta t^2} - \boldsymbol{\Psi}_1\right)\boldsymbol{A}_{1,j}\right. \tag{18}$$

$$\left. + \frac{3}{2}\frac{\boldsymbol{\Psi}_1}{\Delta t} - \boldsymbol{\Psi}_0 - \frac{1}{2}\frac{\boldsymbol{\Psi}_2}{\Delta t^2}\right]\boldsymbol{B}_j, \tag{19}$$

$$\boldsymbol{H}_{j+1-k_\tau} = \left[\left(-\frac{\boldsymbol{\Psi}_3}{\Delta t^2} + \frac{\boldsymbol{\Psi}_2}{\Delta t}\right)\boldsymbol{A}_{1,j} + \frac{\boldsymbol{\Psi}_2}{\Delta t^2} - 2\frac{\boldsymbol{\Psi}_1}{\Delta t}\right]\boldsymbol{B}_j, \tag{20}$$

$$\boldsymbol{H}_{j+2-k_\tau} = \frac{1}{2}\left(\frac{\boldsymbol{\Psi}_1}{\Delta t} - \frac{\boldsymbol{\Psi}_2}{\Delta t^2}\right)\boldsymbol{B}_j. \tag{21}$$

The coefficients $\boldsymbol{\Psi}_i$ are defined as

$$\boldsymbol{\Psi}_0 = \int_0^{\Delta t} \mathrm{e}^{\boldsymbol{A}_0(\Delta t - \xi)}\xi^0 \mathrm{d}\xi = \boldsymbol{A}_0^{-1}(\mathrm{e}^{\boldsymbol{A}_0\Delta t} - \Delta t^0\boldsymbol{I}), \tag{22}$$

$$\boldsymbol{\Psi}_1 = \int_0^{\Delta t} \mathrm{e}^{\boldsymbol{A}_0(\Delta t - \xi)}\xi^1 \mathrm{d}\xi = \boldsymbol{A}_0^{-1}(\boldsymbol{\Psi}_0 - \Delta t^1\boldsymbol{I}), \tag{23}$$

$$\boldsymbol{\Psi}_2 = \int_0^{\Delta t} \mathrm{e}^{\boldsymbol{A}_0(\Delta t-\xi)}\xi^2 \mathrm{d}\xi = \boldsymbol{A}_0^{-1}(2\boldsymbol{\Psi}_1 - \Delta t^2 \boldsymbol{I}), \tag{24}$$

$$\boldsymbol{\Psi}_3 = \int_0^{\Delta t} \mathrm{e}^{\boldsymbol{A}_0(\Delta t-\xi)}\xi^3 \mathrm{d}\xi = \boldsymbol{A}_0^{-1}(3\boldsymbol{\Psi}_2 - \Delta t^3 \boldsymbol{I}). \tag{25}$$

Equation (14) can finally be written as a map:

$$\boldsymbol{\Xi}_{j+1} = \boldsymbol{D}_j \boldsymbol{\Xi}_j. \tag{26}$$

The matrix $\boldsymbol{D}_j$ describes the transition from the beginning to the end of the subinterval. If $\boldsymbol{Q}_j$ is orthogonal, then $\boldsymbol{D}_j$ reads as follows:

$$\boldsymbol{D}_j = \begin{pmatrix} \boldsymbol{Q}_j^{-1}\boldsymbol{H}_j & \boldsymbol{0} \cdots & \boldsymbol{Q}_j^{-1}\boldsymbol{H}_{j-k_\tau+2} & \boldsymbol{Q}_j^{-1}\boldsymbol{H}_{j-k_\tau+1} & \boldsymbol{Q}_j^{-1}\boldsymbol{H}_{j-k_\tau} \\ \boldsymbol{I} & \boldsymbol{0} \cdots & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \vdots & & & & \vdots \\ \boldsymbol{0} & \boldsymbol{0} \cdots & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \end{pmatrix}. \tag{27}$$

The transition matrix $\boldsymbol{\Phi}$ is then obtained by successive multiplication of all the $\boldsymbol{D}_j$ of one system period:

$$\boldsymbol{\Phi} = \boldsymbol{D}_{k_T-1}\boldsymbol{D}_{k_T-2}\cdots\boldsymbol{D}_0. \tag{28}$$

The stability of the system can thus be determined by evaluating criterion (5).

Expressions (15)–(20) can be simplified in certain cases, and this makes it possible to reduce the number of computations. For example, for most cases that arise in machining, the time variation and the time delay are typically introduced by the process force law and the corresponding input and output behavior. Typical process force laws depend only on the current and past displacements of the workpiece-machine-tool system. In this case, a reordering of the states makes it possible to bring the system matrices into the form

$$\boldsymbol{A}_1(t) = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{a}(t) & \boldsymbol{0} \end{bmatrix} \quad \text{and} \quad \boldsymbol{B}(t) = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{b}(t) & \boldsymbol{0} \end{bmatrix}, \tag{29}$$

where $\boldsymbol{a}(t)$ and $\boldsymbol{b}(t)$ are submatrices of appropriate dimension. Therefore, the products $\boldsymbol{A}_{1,j}\boldsymbol{A}_{1,j}$ and $\boldsymbol{A}_{1,j}\boldsymbol{B}_j$ in (15)–(20) vanish. This is also true of the Mathieu equations investigated subsequently in this work. The coefficients of (14) can then be simplified to

$$\boldsymbol{Q}_j = \boldsymbol{I} - \left(\frac{\boldsymbol{\Psi}_3\boldsymbol{A}_0}{\Delta t^2} + 3\frac{\boldsymbol{\Psi}_2}{\Delta t^2} - \frac{\boldsymbol{\Psi}_2\boldsymbol{A}_0}{\Delta t} - 2\frac{\boldsymbol{\Psi}_3}{\Delta t^3}\right)\boldsymbol{A}_{1,j},$$

$$\boldsymbol{H}_j = \left(\boldsymbol{\Psi}_1\boldsymbol{A}_0 + 2\frac{\boldsymbol{\Psi}_3}{\Delta t^3} - 2\frac{\boldsymbol{\Psi}_2\boldsymbol{A}_0}{\Delta t} + \frac{\boldsymbol{\Psi}_3\boldsymbol{A}_0}{\Delta t^2} - 3\frac{\boldsymbol{\Psi}_2}{\Delta t^2} + \boldsymbol{\Psi}_0\right)\boldsymbol{A}_{1,j} + \mathrm{e}^{\boldsymbol{A}_0\Delta t},$$

$$\boldsymbol{H}_{j-k_\tau} = \left(\frac{3}{2}\frac{\boldsymbol{\Psi}_1}{\Delta t} - \boldsymbol{\Psi}_0 - \frac{1}{2}\frac{\boldsymbol{\Psi}_2}{\Delta t^2}\right)\boldsymbol{B}_j,$$

$$\boldsymbol{H}_{j-k_\tau+1} = \left(\frac{\boldsymbol{\Psi}_2}{\Delta t^2} - 2\frac{\boldsymbol{\Psi}_1}{\Delta t}\right)\boldsymbol{B}_j,$$

$$\boldsymbol{H}_{j-k_\tau+2} = \frac{1}{2}\left(\frac{\boldsymbol{\Psi}_1}{\Delta t} - \frac{\boldsymbol{\Psi}_2}{\Delta t^2}\right)\boldsymbol{B}_j.$$

Although the method is presented for a system with a single discrete delay, analysis of systems with multiple or continuous delays is possible using approaches similar to those for other discretization techniques, for example, those described for the SDM in Ref. [21].

3 Numerical results and discussion

The presented analysis procedure is now implemented and tested using two example problems. The results are compared to those obtained by existing methods.

3.1 Damped delayed Mathieu equations

Damped delayed Mathieu equations are among the most widely investigated problems that involve delay. The stability properties are therefore well known, which makes them an excellent benchmark for stability analysis methods.

The damped delayed Mathieu equation has the form [14]

$$\ddot{\boldsymbol{x}}(t) + \kappa\dot{\boldsymbol{x}}(t) + (\delta + \varepsilon\cos(2\,\pi\,t/T))\boldsymbol{x}(t) = b\boldsymbol{x}(t-\tau). \tag{30}$$

By Cauchy transformation, it can be expressed in state-space form with the state vector $\boldsymbol{u} = [\boldsymbol{x}\ \dot{\boldsymbol{x}}]^{\mathrm{T}}$. Separating the time-dependent and delayed parts of the equation gives

$$\dot{\boldsymbol{u}}(t) = (\boldsymbol{A}_0 + \boldsymbol{A}_1(t))\boldsymbol{u}(t) - \boldsymbol{B}\boldsymbol{u}(t-\tau), \tag{31}$$

where

$$\boldsymbol{A}_0 = \begin{pmatrix} 0 & 1 \\ -\delta & -\kappa \end{pmatrix}, \quad \boldsymbol{A}_1(t) = \begin{pmatrix} 0 & 0 \\ -\varepsilon\cos(2\,\pi\,t/T) & 0 \end{pmatrix},$$

$$\boldsymbol{B} = \begin{pmatrix} 0 & 0 \\ -b & 0 \end{pmatrix}. \tag{32}$$

The stability properties of Mathieu equations depend quite strongly on the parameters [2,3]. Consequently, we consider three example cases, where $\tau < T$, $\tau = T$, and $\tau > T$ to show that the proposed method can be successfully applied to a wide variety of Mathieu equations. With $\tau = 2\,\pi$, the cases are

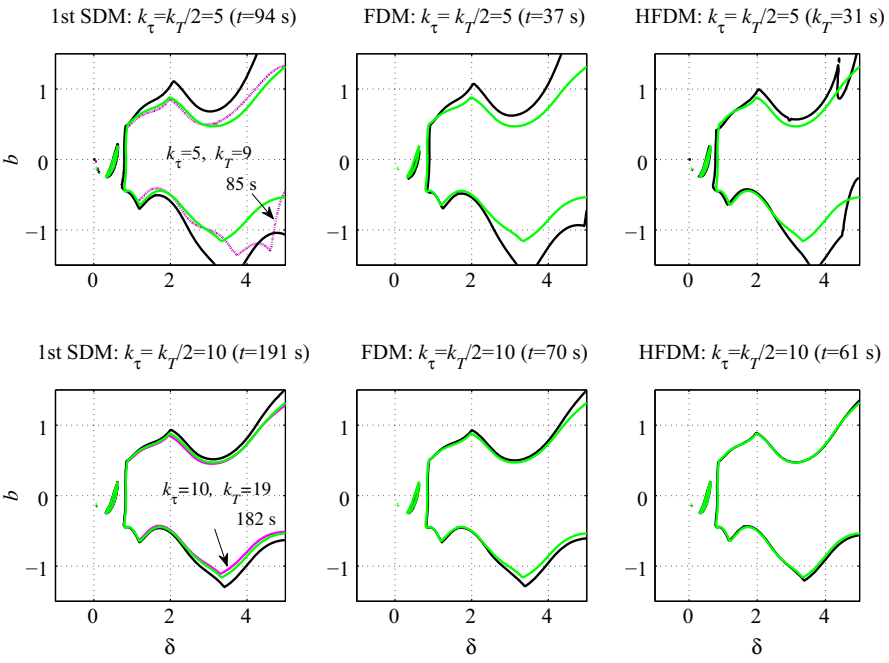

Fig. 2 Stability charts for Mathieu equation (case 1 $T = 4\,\pi, \epsilon = 1, \kappa = 0.2$)

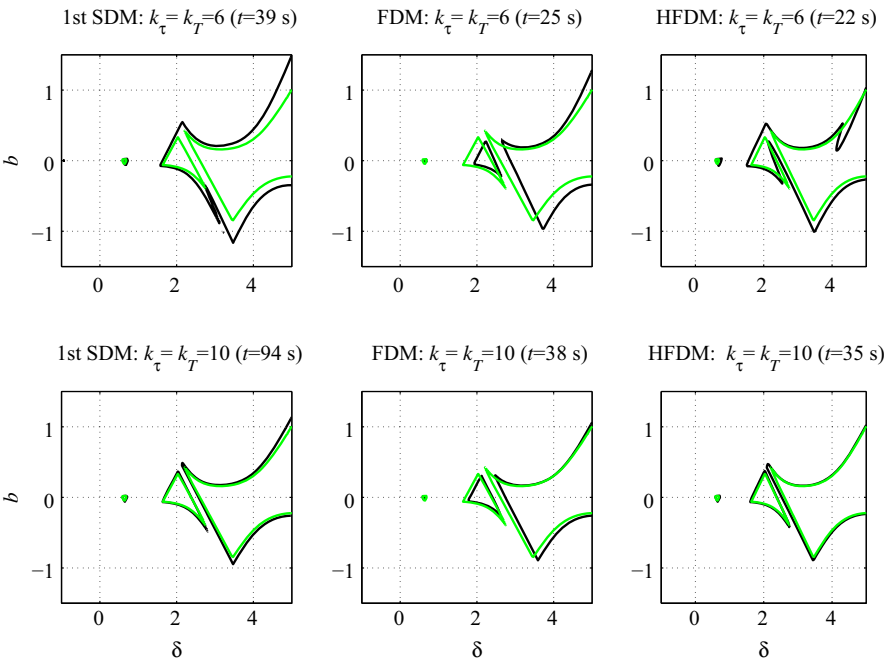

Fig. 3 Stability charts for Mathieu equation (case 2 $T = 2\,\pi, \epsilon = 2, \kappa = 0.1$)

$$\begin{aligned} &\text{case 1: } T = 4\,\pi, \varepsilon = 1, \kappa = 0.2, \\ &\text{case 2: } T = 2\,\pi, \varepsilon = 2, \kappa = 0.1, \\ &\text{case 3: } T = 1\,\pi, \varepsilon = 1, \kappa = 0.2. \end{aligned} \tag{33}$$

The corresponding stability charts are shown in Figs. 2, 3, and 4. For comparison, stability charts constructed using the first-order SDM [15] and the FDM [16] are shown together.

Note that the number of discretization steps k_T and k_τ relate directly to the discretization timestep Δt.

Both approximation parameters k_T and k_τ are important for the final results obtained via the SDM. The relations between k_T and k_τ in the SDM are

$$k_\tau = \text{int}\left(\frac{\tau + \Delta t/2}{\Delta t}\right) = \text{int}\left(k_T \frac{\tau}{T} + \frac{1}{2}\right). \tag{34}$$

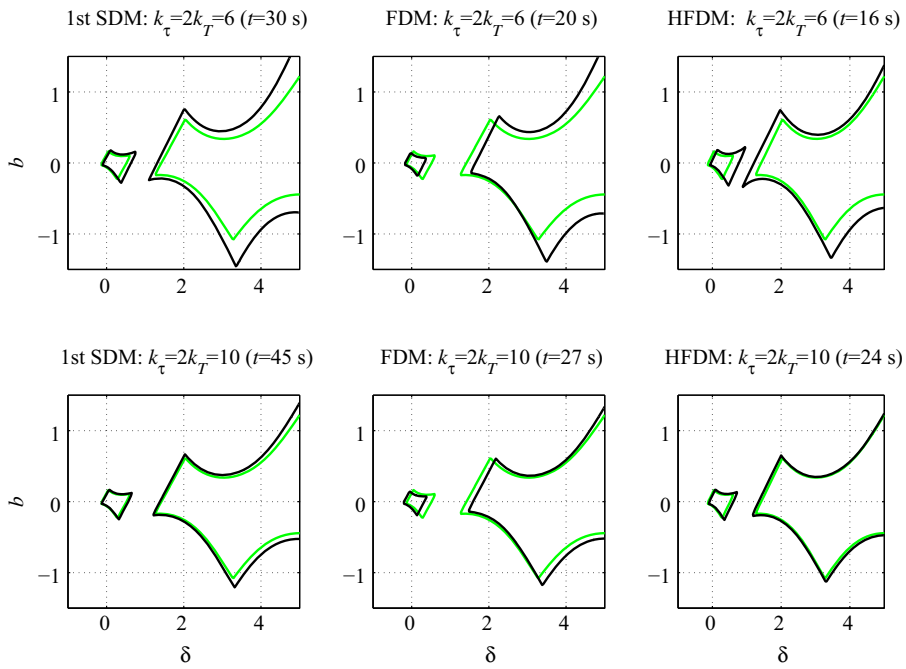

Fig. 4 Stability charts for Mathieu equation (case 3 $T = \pi, \epsilon = 1, \kappa = 0.2$)

For example, if $\tau = 2\,\pi$ and $T = 4\,\pi$, then $k_\tau = 10$ for both $k_T = 19$ and $k_T = 20$. Thus two different charts can be constructed for every k_τ. Contrary to the SDM, there is a one-to-one correspondence between k_T and k_τ in the proposed HFDM.

Figure 2 shows the stability charts for case 1, Fig. 3 for case 2, and Fig. 4 for case 3. The charts were obtained using the proposed HFDM as well as the established SDM and FDM. The green curves are the exact curves, obtained using the first-order SDM and a very large number of 100 discretization steps. Because there is not a one-to-one correspondence between k_τ and k_T, two charts for the SMD are drawn in case 1. There is a visible difference between the two charts constructed for the same k_τ but different k_T. The chart with the higher k_T is closer to the correct result than the other and will be used for comparison with other methods.

Summing up the results, the proposed method yields results that are much closer to the exact solution for the same level of discretization than those obtained using the other methods. The SDM converges faster than the FDM but takes the most computational time.

Table 1 shows the number of discretization steps and the relative computational time needed to obtain a curve that is visually indistinguishable from the precise results. The precise results were computed using the SDM and a large number of discretization steps. The charts were drawn using an equally spaced 200×100 grid of calculated eigenvalues in the shown parameter range. The baseline of 100 % is defined by the time needed by the proposed method. Using a standard desktop PC (DualCore 2.5 GHz, 4 GB RAM), it took 66 s for case 1, 38 s for case 2, and 27 s for case 3. The time advantage of the proposed method is quite impressive – only a third to a quarter of the computational time of the SDM. It economizes approximately an additional one-third of the computational time used by the first-order FDM. It is important to note that the computational time and the number of discretization steps are lower. This is in contrast to the FDM proposed in Ref. [16], which to a certain extent sacrifices convergence for a lower computational time in this case.

Table 1 Steps and computational time needed for different methods to get obtain stability charts

Method	Case 1			Case 2			Case 3		
	k_T	k_τ	Time	k_T	k_τ	Time	k_T	k_τ	Time
First FDM	24	12	156 %	18	18	139 %	12	24	193 %
First SDM	21	11	318 %	14	14	388 %	9	18	296 %
HFDM	20	10	100 %	12	12	100 %	7	14	100 %

To obtain a clearer view of the efficiency of the three methods, convergence rates are investigated for the preceding examples. In Figs. 2, 3, and 4, the largest errors exist around points $(b = -1.1, \delta = 3.4)$, $(b = -0.8, \delta = 3.5)$, and $(b = -0.5, \delta = 4.9)$. Those points lie in regions where larger numbers of discretization steps are necessary to obtain good results. Consequently, the critical eigenvalues corresponding to these points are computed using the three methods.

The results for different discretization levels are shown in Fig. 5. The oscillating behavior of the eigenvalues computed

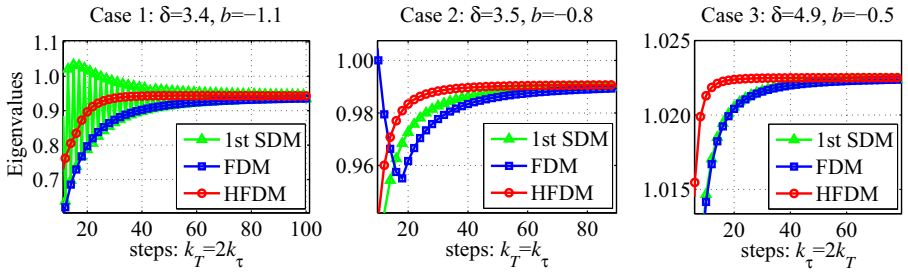

Fig. 5 Convergence of critical eigenvalues

using the SDM for the case $\tau < T$ is interesting to note. The two different possibilities for choosing k_T for each k_τ lead here to two distinct curves with different convergences. Again, the figure illustrates the faster convergence of the proposed method.

3.2 Elastic multibody model of an inside turning operation

In addition to the Mathieu equations presented earlier, we consider now the turning process sketched in Fig. 6. The system was presented and its stability was analyzed in Ref. [26], together with an adaptronic tool holder and different feedback control laws. In Ref. [18], the dynamic stability of different thin-walled and cylindrical workpieces with a rigid tool were analyzed.

The workpiece is cylindrical and hollow with a wall thickness of $w = 1\,\text{mm}$, a length of $l = 0.2\,\text{m}$, and a diameter of $d = 0.12\,\text{m}$. It rotates at a given rotational velocity ω around its symmetry axis. A three-jaw chuck clamps the cylinder and connects to the main spindle of the machine. The material of the workpiece is steel with a Young's modulus of $210 \times 10^9\,\text{N/mm}^2$ and Poisson ratio of 0.3. The process force $\boldsymbol{F}$ acts on both workpiece and tool.

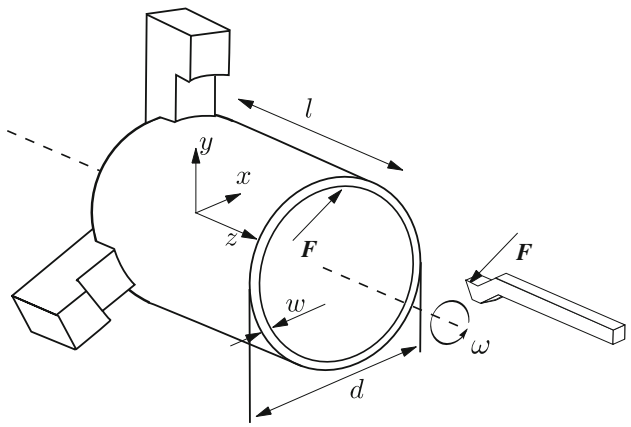

Fig. 6 Schematic view of considered inside turning process

The tool is rather long and slender, which enables it to reach inside the workpiece. Both tool and workpiece must therefore be modeled as flexible bodies. The deformations due to chucking remain in the linear elastic range. Although cutting causes nonlinear effects in the cutting region, the deformations of the overall workpiece remain elastic. Therefore, flexible multibody system theory using a floating frame of reference formulation [22] seems to be well suited to account for the elastic deformations as well as the large nonlinear describable motions of workpiece and tool at the same time.

If the deformations remain linear and elastic, then the equations of motion of an elastic multibody system can be written in the following form [23]:

$$\begin{bmatrix} \boldsymbol{M}_{rr}(\boldsymbol{q}) & \boldsymbol{M}_{rf}(\boldsymbol{q}) \\ \boldsymbol{M}_{fr}(\boldsymbol{q}) & \boldsymbol{M}_{ff} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{r}} \\ \ddot{\boldsymbol{q}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{h}_r(\dot{\boldsymbol{r}}, \boldsymbol{q}, \dot{\boldsymbol{q}}) \\ \boldsymbol{h}_f(\dot{\boldsymbol{r}}, \boldsymbol{q}, \dot{\boldsymbol{q}}) \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} \\ -\boldsymbol{K}_f \boldsymbol{q} - \boldsymbol{D}_f \dot{\boldsymbol{q}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_r \\ \boldsymbol{B}_e \end{bmatrix} \boldsymbol{F}. \tag{35}$$

The rigid body motion is given by the displacements and rotations of a body-fixed reference frame grouped in vector $\boldsymbol{r}$.

Using the finite-element method, the deformations of the elastic bodies are described by a number n_e of nodal displacements $\boldsymbol{q}$ given with respect to the body-fixed frame. A nodal displacement is the displacement of a discrete point in the volume of the body. The submatrices $\boldsymbol{M}_{ff} \in \mathbb{R}^{n_e \times n_e}$ and $\boldsymbol{K}_f \in \mathbb{R}^{n_e \times n_e}$ are respectively the mass and stiffness matrices of the flexible part obtained from finite-element analysis (FEA), $\boldsymbol{D}_f \in \mathbb{R}^{n_e \times n_e}$ is the damping matrix calculated under the assumption of Rayleigh damping, i.e., $\boldsymbol{D} = \alpha \boldsymbol{M} + \beta \boldsymbol{K}$. The matrix $\boldsymbol{M}_{rr} \in \mathbb{R}^{6\times 6}$ corresponds to the mass matrix known from rigid multibody dynamics, and the rotational inertia depends on potential elastic deformations. The matrix $\boldsymbol{M}_{fr} = \boldsymbol{M}_{rf}^{\mathrm{T}} \in \mathbb{R}^{n_e \times 6}$ couples the elastic deformations and the rigid body movement. The vectors $\boldsymbol{h}_r$ and $\boldsymbol{h}_f$ collect generalized inertia forces. The external force $\boldsymbol{F}$ is distributed onto the rigid and elastic DOF by the input matrices $\boldsymbol{B}_r$ and $\boldsymbol{B}_e$. The quantities in Eq. (35) are calculated from FEA results and expanded into a Taylor series expan-

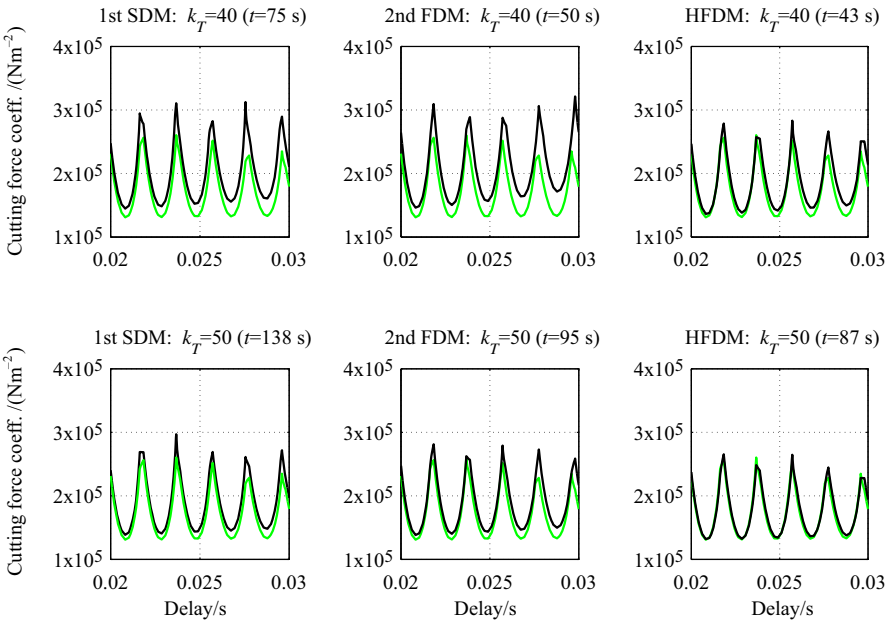

Fig. 7 Stability charts of turning example with different methods

sion. Typically, only constant and linear terms are considered [24]. The terms of order two and higher play only a minor role.

The dimension of the nodal displacement vector $\boldsymbol{q}$ is usually high due to the fine mesh used in FEA. The number of degrees of freedom in Eq. (35) is then in general too high to allow for an efficient numerical implementation. Model order reduction techniques [25] overcome this problem and make it possible to reduce the number of generalized coordinates while maintaining a reasonable accuracy of the results. In this work, modal reduction using the first few eigenmodes of the system is used. This makes it possible to capture the chatter vibrations with sufficient accuracy.

In the case of turning, we consider only one external force, the three-dimensional process force $\boldsymbol{F}$. As the relative position of workpiece and tool changes, the application point of the process force on the inside of the workpiece also changes. Thus the matrices $\boldsymbol{B}_e$ and $\boldsymbol{B}_r$ in Eq. (35) must also change. The inside turning of the thin-walled cylinders indeed exhibits a time-varying input–output behavior [26,27]. We refer the reader to the indicated works for details on how to realize the varying input and output in the considered case.

The workpiece and tool elasticities are dominant in the depth of the cut direction, i.e., perpendicular to the cylinder surface. We therefore neglect the chip thickness variation when modeling the cutting force. The actual depth of a cut depends on the present state of the system and the system past:

$$a_p = \boldsymbol{N}(t)(\boldsymbol{q}(t) - \mu \boldsymbol{q}(t-\tau)). \tag{36}$$

Table 2 Steps and time needed to obtain precise stability charts using different methods

	k_T	k_τ	Time(s)	
Second FDM	70	70	197	270 %
First SDM	70	70	309	423 %
HFDM	55	55	73	100 %

The output matrix $\boldsymbol{N}$ projects the nodal displacements toward the depth of the cut. The factor μ describes the overlapping of successive cuts. We chose it to be one, which represents a worst-case scenario.

What makes this problem interesting, compared to previously analyzed Mathieu equations, is the increased system dimension and the relatively long delay of the system. Both result in a much higher dimension of the transition matrix of the approximated discrete system.

The stability charts of the system for $k_T = k_\tau = 40$ and $k_T = k_\tau = 50$ are shown in Fig. 7. The charts were obtained using the proposed HFDM, the first-order SDM, and a second-order FDM. The charts were drawn using an equally spaced 400 × 200 grid of calculated eigenvalues in the shown parameter range. The precise curve in green was determined using a first-order SDM and a large number – 200 – of discretization steps.

Since the second-order FDM can be used for the turning process here, and it converges faster, we use it instead of the first-order FDM for comparison with the HFDM. Moreover, due to the complexity of the turning model, it requires many more discretization steps, and as a result, the time needed for

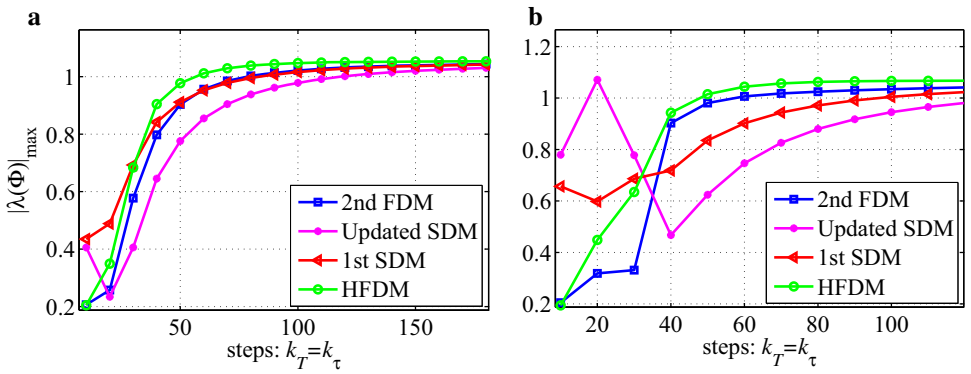

Fig. 8 Convergence of critical eigenvalues for turning model

the matrix multiplication (28) is very high. Therefore, the efficient matrix multiplication method in Ref. [28] is used here for all three methods.

The computational time and discretization steps needed to calculate a stability chart that is visually indistinguishable from an exact one are given in Table 2. The table confirms the previously obtained results. For the considered turning problems with increased system dimension and much larger delay times, the efficiency gains of the approach are much more pronounced than for the Mathieu equation.

Finally, the convergence rates are investigated for the turning example around a valley ($a: \tau = T = 0.0288, k_{cf} = 1.347 \times 10^5$) and a peak value($b: \tau = T = 0.0296, k_{cf} = 2.357 \times 10^5$) in Fig. 8, which also shows that the proposed method converges faster than the other methods.

4 Conclusion

In this paper, an efficient high-order FDM was proposed to analyze the stability of time-periodic delay differential equations. The method discretizes the system using Hermite interpolation for past states and second-order Lagrangian polynomials for the present state of a system. The approach was successfully implemented using two example problems. The delayed Mathieu equation is widely used as a benchmark and makes it possible to compare the performance of the proposed method to a variety of existing solutions. The model of an inside turning operation, on the other hand, is challenging because of comparably long delays and the increased number of states needed to describe the elastic deformation of the thin-walled workpiece.

Using the examples, it could be shown that the proposed HFDM converges against the precise values with an increasing number of discretization steps. Furthermore, it converges faster than the well-established methods and requires less computational time. Even compared to the very efficient second-order FDM, up to half of the computational time can be saved. This is a quite significant gain when large systems and many different configurations are being investigated.

Acknowledgments This project was partially supported by a scholarship from the China Scholarship Council while Y.L. was visiting the University of Stuttgart. A.F. and P.E. would like to thank the German Research Foundation (DFG) for financial support within the Cluster of Excellence in Simulation Technology (EXC 310) at the University of Stuttgart.

References

1. Ruby, L.: Applications of the Mathieu equation. Am. J. Phys. **64**, 39–44 (1996)
2. Insperger, T., Stépán, G.: Stability chart for the delayed Mathieu equation. Proc. R. Soc. Lond. Ser. A **458**, 1989–1998 (2002)
3. Insperger, T., Stépán, G.: Stability of the damped Mathieu equation with time delay. J. Dyn. Syst. Meas. Control **125**, 166–171 (2003)
4. Morrison, T.M., Rand, R.H.: 2:1 resonance in the delayed nonlinear Mathieu equation. Nonlinear Dyn. **50**, 341–352 (2007)
5. Altintas, Y., Engin, S., Budak, E.: Analytical stability prediction and design of variable pitch cutters. J. Manuf. Sci. Eng. **121**, 173–178 (1999)
6. Insperger, T., Stépán, G., Bayly, P.V., et al.: Multiple chatter frequencies in milling processes. J. Sound Vib. **262**, 333–345 (2003)
7. Segalman, D.J., Butcher, E.A.: Suppression of regenerative chatter via impedance modulation. J. Vib. Control **6**, 243–256 (2000)
8. Stépán, G.: Modelling nonlinear regenerative effects in metal cutting. Philos. Trans. R. Soc. Lond. Ser. A **359**, 739–757 (2001)
9. Campbell, S.A., Ruan, S., Wei, J.: Qualitative analysis of a neural network model with multiple time delays. Int. J. Bifurcat. Chaos **9**, 1585–1595 (1999)
10. Insperger, T., Stépán, G.: Stability improvements of robot control by periodic variation of the gain parameters. In: Proceedings of the 11th World Congress in Mechanism and Machine Science, 1816–1820 (2004)
11. Stépán, G.: Retarded Dynamical Systems: Stability and Characteristic Functions. Longman Scientific & Technical Marlow, New York (1989)
12. Stépán, G., Kollár, L.: Balancing with reflex delay. Math. Comput. Model. **31**, 199–205 (2000)
13. Insperger, T., Stépán, G.: Semi-discretization method for delayed systems. Int. J. Numer. Methods Eng. **55**, 503–518 (2002)

14. Insperger, T., Stépán, G.: Updated semi-discretization method for periodic delay-differential equations with discrete delay. Int. J. Numer. Methods Eng. **61**, 117–141 (2004)
15. Insperger, T., Stépán, G., Turi, J.: On the higher-order semi-discretizations for periodic delayed systems. J. Sound Vib. **313**, 334–341 (2008)
16. Ding, Y., Zhu, L., Zhang, X., et al.: A full-discretization method for prediction of milling stability. Int. J. Mach. Tools Manuf. **50**, 502–509 (2010)
17. Liu, Y., Zhang, D., Wu, B.: An efficient full-discretization method for prediction of milling stability. Int. J. Mach. Tools Manuf. **63**, 44–48 (2012)
18. Chanda, A., Fischer, A., Eberhard, P., et al.: Stability analysis of a thin-walled cylinder in turning operation using the semi- discretization method. Acta Mech. Sin. **30**, 214–222 (2013)
19. Fischer, A., Eberhard, P.: Improving the dynamic stability of a workpiece dominated turning process using an adaptronic tool holder. Theor. Appl. Mech. Lett. **3**, 013008 (2013)
20. Hale, J.K., Lunel, S.M.V.: Introduction to Functional Differential Equations. Springer, New York (1993)
21. Insperger, T., Stépán, G.: Semi-Discretization for Time-Delay Systems: Stability and Engineering Applications. Springer, New York (2011)
22. Shabana, A.A.: Dynamics of Multibody Systems. Cambridge University Press, Cambridge (2005)
23. Ambrósio, J.A.C.: Distributed deformation: a finite element method. In: Ambrósio, J.A.C., Eberhard, P. (eds.) Advanced Design of Mechanical Systems: From Analysis to Optimization, pp. 351–374. Springer, Berlin (2009)
24. Wallrapp, O.: Standardization of flexible body modeling in multibody system codes, part I: definition of standard input data. Mech. Struct. Mach. **22**, 283–304 (1994)
25. Fehr, J.: Automated and error controlled model reduction in elastic multibody systems. Schriften aus dem Institut für Technische und Numerische Mechanik der Universität Stuttgart 21, Shaker Verlag, Aachen (2011)
26. Fischer, A., Eberhard, P.: Simulation-based stability analysis of a thin-walled cylinder during turning with improvements using an adaptronic turning chisel. Arch. Mech. Eng. **58**, 367–391 (2011)
27. Fischer, A., Eberhard, P., Ambrósio, J.: Parametric flexible multibody model for material removal during turning. J. Comput. Nonlinear Dyn. **9**, 011007 (2013)
28. Henninger, C., Eberhard, P.: Improving the computational efficiency and accuracy of the semi-discretization method for periodic delay-differential equations. Eur. J. Mech. A/Solids **27**, 975–985 (2008)