Small-Signal Stability Analysis of Large Power Systems with Inclusion of Multiple Delays

F. Milano, *Senior Member, IEEE*

Abstract— **The paper focuses on the small-signal stability analysis of large power systems with inclusion of multiple delayed signals. The following four techniques are compared: (i) a Chebyshev discretization scheme of an equivalent partial differential equations that resembles the original delay differential-algebraic equations (DDAEs); (ii) an approximation of the time integration operator; (iii) a linear multi-step discretization of the DDAEs based on an high-order implicit time-integration scheme; and (iv) the well-known Padé approximants. These techniques are compared using a GPU-based parallel implementation of the Shur method and QR factorization and tested through a real-world transmission system.**

Index Terms— **Time delay, delay differential algebraic equations (DDAE), multiple delays, small-signal stability, Chebyshev discretization, linear multi-step (LMS) methods, Padé approximants.**

I. Introduction

RECENT developments of wide area control schemes, the higher and higher penetration of distributed generations with decentralized controls and the increased number of measurements based on telecommunication systems (e.g., PMUs) lead to an increasing impact of signal delays on power system dynamic response and operation.

This paper focuses on small signal stability analysis of large power systems with inclusion of multiple delays. This analysis involves the formulation of power systems as delayed differential algebraic equations (DDAEs), whose stability analysis pose particularly challenging mathematical and numerical issues.

While time delays are intrinsic of physical and control systems, these are typically neglected or approximated with simple lag blocks in the conventional model of power systems for voltage and transient stability analysis. Most power system devices, e.g., transformers and synchronous machines, are actually not affected by delays, except for very long transmission lines [1], [2]. However, most regulators do and, in recent years, the ubiquitous presence of communication systems and remote measurements, e.g., phasor measurement units (PMUs), has brought the attention on the impact of the issues related to the communication of remote control signals and the impact of the delays of these signals on the stability of the power grid.

The focus of most research papers in this field is, as natural, devoted to the design of robust controllers that are able to reduce the impact of communication delays. The following papers are recent contributions to the robust control of wide area control schemes (WACS) [3]–[10]. The main goal of the papers above is to improve the effect of power system stabilizers to damp interarea oscillations. Another emerging area where delays are relevant is the load frequency control [11], [12].

This paper focuses on the evaluation of the small-signal stability of large DDAEs. Delays transform the classical problem of finding the roots of the state matrix of the system at the equilibrium point into the solution of a transcendental characteristic equation, with *infinitely many roots*. While there are attempts to define an exact analytic solution for oversimplified power system models [13]–[15], an explicit solution cannot be found in general. Other approaches are based on the definition of a Lyapunov function with the well-known difficulties to find such a function [16], [17] or on the solution of a linear matrix inequality (LMI) problem [18], [19], whose computational burden, however, is cumbersome but has become tractable for some applications [5]. Other methods are based on the well-known Padé approximants which allow representing the delay as a set of linear differential equations [12].

This paper considers four different approaches that approximate the solution of the small-signal stability of DDAEs. These are: (i) a Chebyshev discretization of a set of partial differential equations (PDEs) that are equivalent to the original DDAEs [20]–[23]; (ii) a discretization of the time integration operator (TIO) as proposed in [24]–[26]; a linear multi-step (LMS) approximation which has been proposed in [27]–[29] and is implemented in the open-source software tool DDE-BIFTOOL [30]; and the well-known Padé approximants [31].

The Chebyshev discretization has been successfully applied to power systems with a single delay [32], [33] and the Padé approximants have been used in [12] but not considered for the solution of the small-signal stability problem. The other two methods are considered for the first time for the analysis of power systems.

A common characteristics of the techniques above is the high computational burden, which, unfortunately, increases more than linearly with the size of the problem. Hence proper numerical schemes and implementations have to be used. This paper exploits a GPU-based numerical library, namely, MAGMA, that provides an efficient parallel implementation of LAPACK functions and QR factorization for solving the linear eigenvalue problem [34], [35].

The novel contributions of the paper are the following:

1) A comparison of four methods to approximate the spectrum of large power system models modelled as a set of DDAE. These methods are: Chebyshev discretization, approximation of the TIO, LMS approximation and Padé approximants.
2) A comprehensive testing of the GPU-based numerical

Federico Milano is with the School of Electrical, Electronic and Communications Engineering, University College Dublin, Ireland (e-mail: federico.milano@ucd.ie).

library MAGMA and state-of-the-art NVidia GPU card for the solution of very large linear eigenvalue problems.

The remainder of the paper is organized as follows. Section II defines a general model of delayed power systems and introduces the techniques to evaluate the small-signal stability of DDAE with inclusion of multiple delays. Section III presents simulation results based on a dynamic 1479-bus model of the all-island Irish system. Conclusions are drawn in Section IV.

II. Small-signal Stability of Delayed Power Systems Models

This section briefly recalls definitions and outlines the theoretical background on small-signal stability analysis of DDAEs with inclusion of multiple delays and introduces four techniques to approximate the spectrum of DDAEs with inclusion of multiple delays.

A. General Model of Power Systems with inclusions of Delays

The conventional power system model used for solving voltage and transient stability analyses consists of a set of differential algebraic equations (DAEs) as follows [36]:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{u}) \\ \boldsymbol{0} &= \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{u}) \end{aligned} \tag{1}$$

where $\boldsymbol{f}$ ($\boldsymbol{f}: \mathbb{R}^{n+m+p} \mapsto \mathbb{R}^n$) are the differential equations, $\boldsymbol{g}$ ($\boldsymbol{g}: \mathbb{R}^{n+m+p} \mapsto \mathbb{R}^m$) are the algebraic equations, $\boldsymbol{x}$ ($\boldsymbol{x} \in \mathbb{R}^n$) are the state variables, $\boldsymbol{y}$ ($\boldsymbol{y} \in \mathbb{R}^m$) are the algebraic variables, and $\boldsymbol{u}$ ($\boldsymbol{u} \in \mathbb{R}^p$) are discrete variables modeling events, e.g., line outages and faults.

The DDAE formulation is obtained by introducing time delays in (1). Let

$$\begin{aligned} x_d &= x(t-\tau) \\ y_d &= y(t-\tau) \end{aligned} \tag{2}$$

be the *retarded* or *delayed* state and algebraic variables, respectively, where t is the current simulation time, and τ ($\tau > 0$) is the time delay. In the remainder of this paper, since the main focus is on small-signal stability analysis, time delays are assumed to be constant.

If some state and or algebraic variables in (1) are affected by a time delay as in (2), one obtains:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{x}_d, \boldsymbol{y}_d, \boldsymbol{u}) \\ \boldsymbol{0} &= \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{x}_d, \boldsymbol{u}) \end{aligned} \tag{3}$$

which is the index-1 Hessenberg form of DDAE given in [37], [38]. Note that $\boldsymbol{g}$ do not depend on $\boldsymbol{y}_d$. This allows obtaining a closed form for the small-signal stability analysis and, as discussed in [32], (3) is adequate to model, without lack of generality, power system models. Note that this assumption is well satisfied in physical systems such as power system ones as it is quite uncommon that the same delay affects several variables and, in particular, both state and algebraic ones, of the system.

B. Small-Signal Stability of DDAE

Assume that a stationary solution of (3) is known and has the form:

$$\begin{aligned} \boldsymbol{0} &= \boldsymbol{f}(\boldsymbol{x}_0, \boldsymbol{y}_0, \boldsymbol{x}_0, \boldsymbol{y}_0, \boldsymbol{u}_0) \\ \boldsymbol{0} &= \boldsymbol{g}(\boldsymbol{x}_0, \boldsymbol{y}_0, \boldsymbol{x}_0, \boldsymbol{u}_0) \end{aligned} \tag{4}$$

where it has been used the fact that, in steady-state, $\boldsymbol{x}_{d0} = \boldsymbol{x}_0$ and $\boldsymbol{y}_{d0} = \boldsymbol{y}_0$. Moreover, discrete variables $\boldsymbol{u}_0$ are assumed to be constant in the remainder of this paper. Then, differentiating (3) at the stationary solution yields:

$$\Delta\dot{\boldsymbol{x}} = \boldsymbol{f}_{\boldsymbol{x}}\Delta\boldsymbol{x} + \boldsymbol{f}_{\boldsymbol{x}_d}\Delta\boldsymbol{x}_d + \boldsymbol{f}_{\boldsymbol{y}}\Delta\boldsymbol{y} + \boldsymbol{f}_{\boldsymbol{y}_d}\Delta\boldsymbol{y}_d \tag{5}$$

$$\boldsymbol{0} = \boldsymbol{g}_{\boldsymbol{x}}\Delta\boldsymbol{x} + \boldsymbol{g}_{\boldsymbol{x}_d}\Delta\boldsymbol{x}_d + \boldsymbol{g}_{\boldsymbol{y}}\Delta\boldsymbol{y} \tag{6}$$

where, neglecting without loss of generality singularity-induced bifurcation points, it can be assumed that $\boldsymbol{g}_{\boldsymbol{y}}$ is non-singular. Substituting (6) into (5), one obtains:[1]

$$\Delta\dot{\boldsymbol{x}} = \boldsymbol{A}_0\Delta\boldsymbol{x} + \boldsymbol{A}_1\Delta\boldsymbol{x}(t-\tau) + \boldsymbol{A}_2\Delta\boldsymbol{x}(t-2\tau) \tag{7}$$

where:

$$\boldsymbol{A}_0 = \boldsymbol{f}_{\boldsymbol{x}} - \boldsymbol{f}_{\boldsymbol{y}}\boldsymbol{g}_{\boldsymbol{y}}^{-1}\boldsymbol{g}_{\boldsymbol{x}} \tag{8}$$

$$\boldsymbol{A}_1 = \boldsymbol{f}_{\boldsymbol{x}_d} - \boldsymbol{f}_{\boldsymbol{y}}\boldsymbol{g}_{\boldsymbol{y}}^{-1}\boldsymbol{g}_{\boldsymbol{x}_d} - \boldsymbol{f}_{\boldsymbol{y}_d}\boldsymbol{g}_{\boldsymbol{y}}^{-1}\boldsymbol{g}_{\boldsymbol{x}} \tag{9}$$

$$\boldsymbol{A}_2 = -\boldsymbol{f}_{\boldsymbol{y}_d}\boldsymbol{g}_{\boldsymbol{y}}^{-1}\boldsymbol{g}_{\boldsymbol{x}_d} \tag{10}$$

The first matrix $\boldsymbol{A}_0$ is the well-known state matrix that is computed for standard DAEs of the form (1). The other two matrices are not null only if the system is of retarded type. The matrix $\boldsymbol{A}_1$ is found in any delay differential equations, while $\boldsymbol{A}_2$ appears specifically in DDAEs, although it can be null if either $\boldsymbol{f}$ does not depend on $\boldsymbol{y}_d$ or $\boldsymbol{g}$ does not depend on $\boldsymbol{x}_d$. If one of the two conditions above are satisfied, (14) becomes:

$$\Delta(\lambda) = \lambda\boldsymbol{I}_n - \boldsymbol{A}_0 - \boldsymbol{A}_1 e^{-\lambda\tau} \ , \tag{11}$$

which is the case considered in [32]. Note that, in this paper, no simplification is imposed to (3).

Equation (7) is a particular case of the standard variational form of the linear delay differential equations:

$$\Delta\dot{\boldsymbol{x}} = \boldsymbol{A}_0\Delta\boldsymbol{x}(t) + \sum_{i=1}^{\nu}\boldsymbol{A}_i\Delta\boldsymbol{x}(t-\tau_i) \ . \tag{12}$$

The substitution of a sample solution of the form $e^{\lambda t}\boldsymbol{v}$, with $\boldsymbol{v}$ a non-trivial possibly complex vector of order n, leads to the *characteristic equation* of (12):

$$\det\Delta(\lambda) = 0 \tag{13}$$

where

$$\Delta(\lambda) = \lambda\boldsymbol{I}_n - \boldsymbol{A}_0 - \sum_{i=1}^{\nu}\boldsymbol{A}_i e^{-\lambda\tau_i} \tag{14}$$

is called the *characteristic matrix* [39]. In (14), $\boldsymbol{I}_n$ is the identity matrix of order n. The solutions of (14) are called the *characteristic roots* or *spectrum*, similar to the finite-dimensional case (i.e., the case for which $\boldsymbol{A}_i = \boldsymbol{0} \ \forall i = 1, \dots, \nu$).

[1] The interested reader can find in [32] the details on how to determine (8)-(10) from (5) and (6).

As for the finite-dimensional case (i.e., $\nu = 0$), the stability of (12) can be defined based on the sign of the roots of (14), i.e., the stationary point is stable if all roots have negative real part, and unstable if there exists at least one eigenvalue with positive real part.

Equation (14) is transcendental and, hence, shows infinitely many roots. In general, the explicit solution of (14) is not known and only approximated numerical solutions of a subset of the roots of (14) can be found, as discussed in the following subsections.

C. Chebyshev discretization scheme

This approach consists in transforming the original problem of computing the roots of a retarded functional differential equations into a matrix eigenvalue problem of a PDE system of infinite dimensions. No loss of information is involved in this step. Then the dimension of the PDE is made tractable using a discretization based on a finite element method.

Consider the single-delay case first. Let $\boldsymbol{D}_N$ be the Chebyshev differentiation matrix of order N (see the Appendix for details) and define

$$\boldsymbol{M} = \left[\begin{array}{ccccc} & & \hat{\boldsymbol{C}} \otimes \boldsymbol{I}_n & & \\ \hline \boldsymbol{A}_1 & \boldsymbol{0} & \dots & \boldsymbol{0} & \boldsymbol{A}_0 \end{array} \right] , \qquad (15)$$

where $\otimes$ indicate the *tensor product* or Kronecker product;[2] $\boldsymbol{I}_n$ is the identity matrix of order n; and $\hat{\boldsymbol{C}}$ is a matrix composed of the first $N-1$ rows of $\boldsymbol{C}$ defined as follows:

$$\boldsymbol{C} = -2\boldsymbol{D}_N/\tau \ . \qquad (16)$$

Then, the eigenvalues of $\boldsymbol{M}$ are an approximated spectrum of (11). As it can be expected, the number of points N of the grid affects the precision and the computational burden of the method, as it is further discussed in the case study.

The matrix $\boldsymbol{M}$ is the discretization of a set of PDEs where the continuum is represented by the interval $\xi \in [-\tau, 0]$. The continuum is discretized along a grid of N points and the position of such points are defined by the Chebyshev polynomial interpolation. The last n rows of $\boldsymbol{M}$ impose the boundary conditions $\xi = -\tau$ (i.e., $\boldsymbol{A}_1$) and $\xi = 0$ (i.e., $\boldsymbol{A}_0$), respectively.

Figure 1 illustrates matrix (15) through a pictorial representation. Each element of the grid is a $n \times n$ matrix and there are N^2 elements. Light gray blocks are defined by the Chebyshev discretization and are very sparse. Dark gray blocks represent the state matrix $\boldsymbol{A}_0$ and delayed matrix $\boldsymbol{A}_1$ that appear in (11). Finally, white blocks indicate null matrices.

Let now consider the general multi-delay case of the characteristic equation (14) and, thus, let assume that there are ν delays, with $\tau_1 < \tau_2 < \cdots < \tau_{\nu-1} < \tau_\nu$. Each point of the Chebyshev grid corresponds to a delay $\theta_k = (N-k)\Delta\tau$, with $k = 1, 2, \dots, N$ and $\Delta\tau = \tau_\nu/(N-1)$. Hence, $k = 1$ corresponds to the state matrix $\boldsymbol{A}_\nu$, which corresponds to the maximum delay τ_ν; and $k = N$ is taken by the non-delayed state matrix $\boldsymbol{A}_0$. If a delay $\tau_i = \theta_k$ for some $k = 2, \dots, N-1$,

[2]See http://www.encyclopediaofmath.org/index.php/Tensor_product for a formal definition of the Kronecker product.

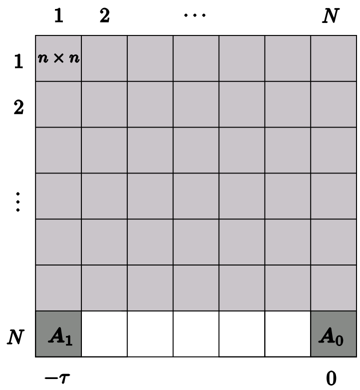

Fig. 1. Representation of the matrix $\boldsymbol{M}$ for a system with a single delay τ and characteristic equation (11).

then the correspondent matrix $\boldsymbol{A}_i$ takes the position k in the grid. Of course, in general, the delays of the system will not match the points of the grid. With this aim, a linear interpolation is considered in this paper, as follows. Let the time delay τ_i, $i \neq k$, satisfy the condition:

$$\theta_k < \tau_i < \theta_{k+1} \ . \qquad (17)$$

Then, the matrices that will be added to the positions k and $k+1$ are, respectively:

$$\boldsymbol{A}_{k,i} = \frac{\tau_i - \theta_k}{\Delta\tau} \boldsymbol{A}_i \ , \quad \boldsymbol{A}_{k+1,i} = \frac{\theta_{k+1} - \tau_i}{\Delta\tau} \boldsymbol{A}_i \ . \qquad (18)$$

Then, the resulting matrix of each point k of the grid is computed as the sum of the contributions of each delay that overlaps that point:

$$\boldsymbol{A}_k = \sum_{i \in \Omega_i} \boldsymbol{A}_{k,i} \ , \qquad (19)$$

where Ω_i is the set of delays τ_i that satisfies (17). Other more sophisticated interpolation schemes can be used. For example, a Lagrange polynomial interpolation is implemented in [25]. Figure 2 illustrates the Chebyshev discretization approach for the multiple-delay case.

D. Discretization of the Time Integration Operator (TIO)

The discretization of the time integration operator that is proposed in [25] is similar to the approach above, but instead of defining the discretization of a PDE, it discretizes directly the set of original DDE equations. For clarity, consider first the single-delay case and consider the following system:

$$\Delta\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_0 \Delta\boldsymbol{x}(t) + \boldsymbol{A}_1 \Delta\boldsymbol{x}(t-\tau) \ , \qquad (20)$$

which is obtained from (7) by assuming that $\boldsymbol{A}_2 = \boldsymbol{0}$. The approach consists in: (i) dividing the interval $[-\tau, 0]$ into a mesh of N intervals with constant step size $h = \tau/N$; and (ii) applying an integration scheme (e.g., a RK method) to the mesh that approximate the continuous solution of (20). Then the discrete counterpart of (20) is given by:

$$\boldsymbol{z}^{i+1} = \boldsymbol{S}_N \boldsymbol{z}^i \ , \qquad (21)$$

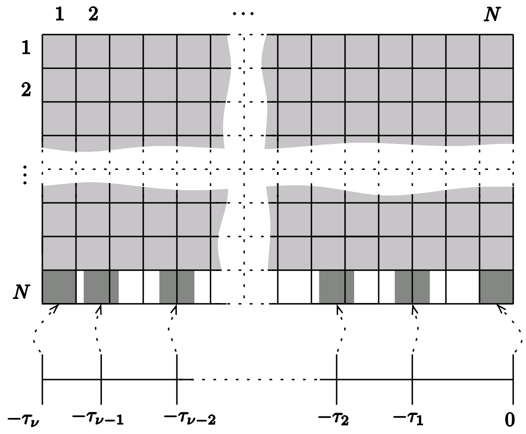

Fig. 2. Representation of the Chebyshev discretization for a system with ν delays $\tau_1 < \tau_2 < \cdots < \tau_{\nu-1} < \tau_\nu$. In the general case, the delays do not match exactly the grid and, thus, an interpolation between consecutive points of the grid is required.

where $\boldsymbol{z} \in \mathbb{R}^{n \cdot r \cdot N}$, and $\boldsymbol{S}_N$ is the following $n \cdot r \cdot N \times n \cdot r \cdot N$ matrix:

$$\boldsymbol{S}_N = \begin{bmatrix} \boldsymbol{B}_0 & \boldsymbol{0} & \dots & \boldsymbol{0} & \boldsymbol{B}_1 \\ \boldsymbol{I}_{nr} & \boldsymbol{0} & \dots & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I}_{nr} & \dots & \boldsymbol{0} & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \dots & \boldsymbol{I}_{nr} & \boldsymbol{0} \end{bmatrix} , \tag{22}$$

where

$$\boldsymbol{B}_0 = \boldsymbol{R} \cdot (\boldsymbol{1}_r \boldsymbol{e}_r^T \otimes \boldsymbol{I}_n) \, , \tag{23}$$
$$\boldsymbol{B}_1 = h\boldsymbol{R} \cdot (\mathcal{A} \otimes \boldsymbol{A}_1) \, , \tag{24}$$

and

$$\begin{aligned} \boldsymbol{R} &= (\boldsymbol{I}_{nr} - h\mathcal{A} \otimes \boldsymbol{A}_0)^{-1} \, , \\ \boldsymbol{1}_r &= (1, \dots, 1)^T \, , \\ \boldsymbol{e}_r &= (0, \dots, 0, 1)^T \, , \end{aligned}$$

and $\mathcal{A}$ is the matrix of the Butcher's tableau that defines the integration scheme, as follows:

$$\begin{array}{c|c} \mathcal{C} & \mathcal{A} \\ \hline & \mathcal{B} \end{array} = \begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \dots & a_{1r} \\ c_2 & a_{21} & a_{22} & \dots & a_{2r} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_r & a_{r1} & a_{r2} & \dots & a_{rr} \\ \hline & b_1 & b_2 & \dots & b_r \end{array} \tag{25}$$

and $\boldsymbol{I}_{nr}$ is the identity matrix of order $n \cdot r$. Note that $\mathcal{A}$ must be invertible, which means that an implicit scheme has to be used (e.g., BDF formulae and Radau methods).

The single-delay case can be extended to the multi-delay one by modifying the first row of the matrix $\boldsymbol{S}_N$ in (22). Let's assume that there are ν delays, with $\tau_1 < \tau_2 < \cdots < \tau_{\nu-1} < \tau_\nu$. Then, the first and the last elements of the first row of (22) are occupied by $\boldsymbol{B}_0$ and $\boldsymbol{B}_\nu$, where $\boldsymbol{B}_0$ is defined as in (23) and $\boldsymbol{B}_\nu$ is:

$$\boldsymbol{B}_\nu = h\boldsymbol{R}(h\boldsymbol{A}_0)(\mathcal{A} \otimes \boldsymbol{A}_\nu) \, . \tag{26}$$

The state matrices associated with the remaining $\nu - 1$ delays are fitted to the grid through a linear interpolation similar to that described in Subsection II-C. The interested reader can find in [25] a more general interpolation approach based on Lagrange polynomials and a detailed discussion on the convergence properties of this LMS discretization approach.

E. Linear Multi-Step (LMS) Approximation

Another possible discretization based on a linear multi-step approximation is that proposed in [29] and implemented in the software tool DDE-BIFTOOL [28]. The time integration operator is discretized using a LMS method with polynomial interpolation to evaluate the delayed terms. Applying a k-step LMS method to (12), one obtains:

$$\sum_{j=0}^{k} \alpha_j \boldsymbol{x}_{L+j} = h \sum_{j=0}^{k} \left[\beta_j \boldsymbol{A}_0 \boldsymbol{x}_{L+j} + \sum_{i=0}^{\nu} \left(\boldsymbol{A}_i \tilde{\boldsymbol{x}}(t_{L+j} - \tau_i) \right) \right] \tag{27}$$

where α_j and β_j are the coefficients of the LMS method and $\tilde{\boldsymbol{x}}(t_{L+j} - \tau_i)$ are approximations of the values of the state variables in past. These are computed using the Nordsieck interpolation, as follows:

$$\tilde{\boldsymbol{x}}(t_p - \epsilon h) = \sum_{\ell=-\rho}^{\sigma} P_\ell(\epsilon) \boldsymbol{x}_{p+\ell}, \quad \epsilon \in [0, 1) \tag{28}$$

where

$$P_\ell = \prod_{k=-\rho, k \neq \ell}^{\sigma} \frac{\epsilon - k}{\ell - k} \tag{29}$$

The resulting method is explicit whenever $\beta_0 = 0$ and $\min\{\tau_i\} > \sigma h$. Further details on this technique can be found in the DDE-BIFTOOL documentation and source code [30].

The LMS-method forms an approximation of the time integration operator over the time step h, hence the roots $\boldsymbol{\mu}$ of the Jacobian matrix of (27) are an approximation of the exponential transforms of the roots $\boldsymbol{\lambda}$ of (14):

$$\boldsymbol{\mu} = \exp(h\boldsymbol{\lambda}) \tag{30}$$

The size of the resulting eigenvalue problem is inversely proportional to the step length h used in the discretization. The choice of h is heuristic and is a critical aspect of this technique. If the step length is too small, the size, say K, of the problem can be huge, e.g., $K \gg n$; if h is too large the approximation of the roots of (14) might not be accurate. The heuristic method to estimate h described in [29] leads to precise results although it is quite conservative. Larger values of h can be obtained using the approach given in more recent works, e.g., [40]. Note also that the procedure given in [29] may lead to determine a high number of extraneous positive roots. A root is discarded if the following condition is satisfied:

$$\text{abs}(\mu_j) > \exp(h \cdot \max\{\tau_i\}), \quad j = 1, 2, \dots, K \tag{31}$$

F. Padé Approximants

The well-known *time shifting* property of the Laplace transform is as follows:

$$f(t-\tau)\,u(t-\tau) \quad \xrightarrow{\mathcal{L}} \quad e^{-\tau s}F(s) \tag{32}$$

where s is the variable of the Laplace transform $\mathcal{L}$, or *complex frequency*; $u(t)$ is the unit step function; and $F(s)$ is the Laplace transform of the function $f(t)$. The approach based on Padé approximants consists in defining a rational polynomial transfer function, say $P(s)$, that approximates $e^{-\tau s}$. Then, the inverse Laplace transform $\mathcal{L}^{-1}$ allows obtaining the approximated time domain function $\phi(t)$ that leads to an approximated DAE as in (1):

$$e^{-\tau s}F(s) \quad \approx \quad P(s)F(s) \quad \xrightarrow{\mathcal{L}^{-1}} \quad \phi(t) \tag{33}$$

Padé approximants are based on the Taylor's expansion of $e^{-\tau s}$ in the frequency domain:

$$\begin{aligned} e^{-\tau s} &= 1 - \tau s + \frac{(\tau s)^2}{2!} - \frac{(\tau s)^3}{3!} + \dots \\ &\approx \frac{b_0 + b_1 \tau s + \dots + b_q (\tau s)^q}{a_0 + a_1 \tau s + \dots + a_p (\tau s)^p}\ , \end{aligned} \tag{34}$$

where coefficients $a_1, \dots, a_p$ and $b_1, \dots, b_q$ are obtained by dividing the polynomials of the right hand side of (34) and imposing that the first $p+q$ coefficients are the same as those of the Taylor's expansion [31]. Note that s has a different meaning than λ in (14). In fact, λ takes an infinite number of discrete values that solve (14), while s is the continuous independent variable of the Laplace transform.

Generally, $p \geq q$ is imposed in (34). If $p = q$, the coefficients a_i and b_i are obtained by the following iterative formula:

$$\begin{aligned} &a_0 = 1\ , \quad a_i \ = \ a_{i-1}\,\frac{p-i+1}{i \cdot (2p-i+1)}\ , \quad \text{and} \\ &b_i = (-1)^i \cdot a_i\ . \end{aligned} \tag{35}$$

The case $p = q$ is noteworthy as the amplitude of the frequency response of the Padé approximant is exact, only the phase is affected by an error. $p = q = 6$ is a common choice in numerical simulations.

The higher the order of the Padé approximant, the lower the phase error (see, for example the discussion on Padé approximants in [12]). Note that, for *small* delays, e.g., of the order of milliseconds, which are common in power systems, the order p of the Padé approximant cannot be too high as the polynomial coefficients depend on the powers of τ.

For example, let $p = 9$ and $\tau = 10^{-3}$ s. Then, one obtains $a_9 = -b_9 = 5.6679 \cdot 10^{-11}$ and $\tau^9 = 10^{-27}$, which leads to $a_9 \cdot \tau^9 = 5.6679 \cdot 10^{-38}$. This number is critically close to the minimum positive value that can be represented by the single-precision binary floating-point defined by the IEEE 754 standard, i.e., $2^{-126} \approx 1.18 \cdot 10^{-38}$. High order Padé approximants may also show unstable poles of *defects* (i.e., a pair of a pole and a zero that are very close but not equal, see [31]). Hence, the floating point representation binds the maximum value of p, being $p^{\max} = q^{\max} = 10$ the most commonly used upper limit.

It is important to note that the approach based on Padé approximants does not deal with (14) but, rather, consists in transforming the set of DDAE into a set of DAE where delays are approximated by a set of linear ordinary differential equations. For the sake of example, let's consider the case of the unit step function $u(t)$. Based on (34), which is in the frequency domain, one can readily obtain the equivalent time-domain function and include it in any standard simulation tool. Consider, for simplicity but without lack of generality, the case $p = q$. The approximant u_d of order p, in time domain, of $u(t-\tau)$ given by (35) is as follows:

$$u_d = \tilde{x}_1 + b_1 \tau \tilde{x}_2 + \dots + b_{p-1}\tau^{p-1}\tilde{x}_p + b_p \tau^p \dot{\tilde{x}}_p \tag{36}$$

where:

$$\dot{\tilde{x}}_i = \tilde{x}_{i+1}, \quad i = 1, 2, \dots, p-1 \tag{37}$$

and

$$a_p \tau^p \dot{\tilde{x}}_p = u - (a_0 \tilde{x}_1 + a_1 \tau \tilde{x}_2 + \dots + a_{p-1}\tau^{p-1}\tilde{x}_p) \tag{38}$$

The set of equations (36)-(38) is linear and introduces p state variables per each delay. Clearly, there is no limitation to the number of delays that can be included in the systems, and there is no structural difference between the single-delay and the multiple-delay case.

III. Case Study

The techniques presented above work satisfactorily for *small* size systems, e.g., few tens of state variables and few tens of delays. The author has tested the four techniques discussed in the previous section on several IEEE benchmark systems, e.g., the IEEE 14 bus system and the IEEE 39 bus system. In all cases, the results obtained with the four techniques for small- and medium-size power systems are very similar. The Chebyshev discretization and the Padé approximants always provide good results for small systems. TIO is also accurate provided that N is increased with respect to the Chebyshev method. For example, $N = 7$ is acceptable for small power systems. On the other hand, LMS provides good results if h is relatively small. For example, $h = 0.1$ s appears acceptable for small power systems. However, standard benchmarks are too small to allow drawing sensible conclusions on the robustness and the accuracy of the techniques discussed Section II for *large* scale eigenvalue problems. Eigenvalue stiffness and the numerical rounding errors play a crucial role as the size of the problem scales up, as shown in this section.

In this case study, the techniques in Subsections II-C to II-F are compared through a dynamic model of the all-island Irish transmission system set up at the UCD Electricity Research Centre. The model includes $1,479$ buses, $1,851$ transmission lines and transformers, 245 loads, 22 conventional synchronous power plants with AVRs and turbine governors, 6 PSSs and 176 wind power plants. The topology and the data of the transmission system are based on the actual real-world system provided by the Irish TSO, EirGrid, but dynamic data are guessed and based on the knowledge of the technology of power plants.

Since the objective is to compare different methods for the small-signal stability analysis of DDAEs, constant time delays

TABLE I

RANGES OF TIME DELAYS INCLUDED IN THE ALL-ISLAND IRISH SYSTEM

Device	Delayed Signal	Delay	Range [s]
Primary voltage regulator	bus voltage	$\tau_{\rm AVR}$	$(0.005, 0.015)$
Power system stabilizer	frequency	$\tau_{\rm PSS}$	$(0.05, 0.25)$
Reheater of steam turbines	steam flow	$\tau_{\rm TG}$	$(3, 11)$
Wind turbine freq. reg.	frequency	$\tau_{\rm TFR}$	$(0.05, 0.25)$
Therm. controlled load	frequency	$\tau_{\rm TCL}$	$(0.05, 0.25)$

TABLE II

NUMBER OF VARIABLES FOR THE ALL-ISLAND IRISH SYSTEM

Model	Type	State vars	Algeb. vars
No delays	DAE	$2,239$	$7,478$
Constant delays	DDAE	$1,935$	$7,338$
Padé approx. ($p = q = 6$)	DAE	$3,415$	$7,929$
Padé approx. ($p = q = 10$)	DAE	$4,399$	$7,929$

are included in most regulators, as follows. All bus terminal voltage measurements of the automatic voltage regulators (AVRs) of the synchronous machines include delays in the range $\tau_{\rm AVR} \in (5, 15)$ ms [32]. The input frequency signal of PSS devices is delayed in the range $\tau_{\rm PSS} \in (50, 250)$ ms [4]. The reheater of the turbine governors of thermal power plants is modelled as a pure delay in the range $\tau_{\rm RH} \in (3, 11)$ s [41].

The model of some variable-speed wind turbines includes a frequency regulation that receives as input the frequency of the center of inertia of the system. The model of the frequency regulator is based on the transient frequency control described in [42]. The frequency signal is assumed to be similar to those of PSS devices, hence $\tau_{\rm TFC} \in (50, 250)$ ms.

Finally, 20% of the loads are assumed to provide a frequency regulation. In other words, 20% of loads are assumed to be equivalent thermostatically controlled heating systems. The dynamic model of these loads and their control is based on [43] and [44], respectively. Again, the input frequency signal is delayed and, in analogy with PSS devices, delays are chosen in the interval $\tau_{\rm TCL} \in (50, 250)$ ms.

The delay ranges considered in this case study are summarized in Table I. In total, the system contains 296 delays ranging in the interval $(0.005, 11)$ s. This wide range is chosen with the purpose of determining the accuracy and the performance of the methods presented in Section II. The resulting DDAE are *stiff* in terms of both device and regulator time constants, which span a range from tens of milliseconds to tens of seconds, and pure time delays.

The order of the system, i.e., the number of state and algebraic variables, depends on the model. Table II shows system statistics for four different models, namely, no delay; constant delays; Padé approximant with $p = q = 6$; and Padé approximant with $p = q = 10$. The only DDAE is the model where delays are implemented as in (3), as Padé approximants transform the delays into a set of linear differential equations.

It is noteworthy that the DDAE is also the model with the lower number of variables. This is due to the fact that, in the standard model with no delays, delays are actually modelled as a simple lag transfer function, each of which introduces a state variable. Note also that, the lag transfer function is, in turn, the Padé approximant with $p = 1$ and $q = 0$. Higher order Padé approximants lead to a substantial increase of the order of the system, and hence of the computational burden of the initialization of system variables and time domain simulations.[3] Transient analysis is out of the scope of this paper but the latter remark has to be kept in mind when choosing the power system models.

All simulations are obtained using Dome, a Python-based power system analysis toolbox [45]. The Dome version used for in this case study is based on Python 3.4.1 (`http://www.python.org`), NVidia Cuda 7.0, Numpy 1.8.2 (`http://numpy.scipy.org`), CVXOPT 1.1.7 (`http://abel.ee.ucla.edu/cvxopt/`), MAGMA 1.6.1 (`icl.cs.utk.edu/magma/software`), and has been executed on a 64-bit Linux Fedora 21 operating system running on a two Intel Xeon 10 Core 2.2 GHz CPUs, 64 GB of RAM, and a 64-bit NVidia Tesla K20X GPU.

Table III shows the 20 rightmost eigenvalues for the all-island Irish transmission system using different system models and techniques. For reference, the first column also shows the 20 rightmost eigenvalues of the non-delayed model. This system does not show any poorly damped mode, i.e., a mode whose damping is below 5%. Column 2 to 5 of Table III show the results obtained using the Chebyshev discretization, the discretization of the time integrator operator (TIO), the linear multi-step (LMS) approximation and the Padé approximants. Two cases are shown for the latter, namely, $p = q = 6$ and $p = q = 10$. Both Chebyshev and TIO discretizations use a grid of order $N = 7$. This number is considered a good trade-off between accuracy and computational burden. The interested reader can find further details on the accuracy of the Chebyshev and TIO discretizations in [32] and [25], respectively. For the discretization of the TIO, a fifth order Radau IIA method is used, with $r = 3$ and the following Butcher's tableau:

$$
\begin{array}{c|ccc}
\frac{2}{5} - \frac{\sqrt{6}}{10} & \frac{11}{45} - \frac{7\sqrt{6}}{360} & \frac{37}{225} - \frac{169\sqrt{6}}{1800} & -\frac{2}{225} + \frac{\sqrt{6}}{75} \\
\frac{2}{5} + \frac{\sqrt{6}}{10} & \frac{37}{225} + \frac{169\sqrt{6}}{1800} & \frac{11}{45} + \frac{7\sqrt{6}}{360} & -\frac{2}{225} - \frac{\sqrt{6}}{75} \\
1 & \frac{4}{9} - \frac{\sqrt{6}}{36} & \frac{4}{9} + \frac{\sqrt{6}}{36} & \frac{1}{9} \\
\hline
 & \frac{4}{9} - \frac{\sqrt{6}}{36} & \frac{4}{9} + \frac{\sqrt{6}}{36} & \frac{1}{9}
\end{array}
\tag{39}
$$

Finally, an Adams-Bashforth 6th order method is used for the LMS approximation, with following coefficients: $\boldsymbol{\alpha} = [1,\ -1,\ 0,\ 0,\ 0,\ 0]$ and $\boldsymbol{\beta} = [0,\ 1901/720,\ -1387/360,\ 109/30,\ -637/360,\ 251/720]$. A time step $h = 0.2$ s is used for the LMS approximation.

To complete the comparison of the four techniques whose results are provided in Table III, Table IV shows the computational burden of these techniques using the GPU-based MAGMA library. The information given in Table IV is the time required to setup the full matrix for the eigenvalue analysis, the order of this matrix, and the time required to solve the full linear eigenvalue problem (LEP).

While all methods are necessarily approximated, the most accurate method to estimate the spectrum of the DDAE can

[3] Padé approximants also lead to increase the number of algebraic variables because the output u_d of the approximated transfer function (34) is algebraic, as shown by (36).

TABLE III

20 RIGHTMOST EIGENVALUES FOR THE ALL-ISLAND IRISH SYSTEM – ALL-DELAY SCENARIO

No delay	Chebyshev Discr. $N = 7$	Discr. of TIO $N \cdot r = 21$	LMS Approx. $h = 0.2$ s	Padé Approx. $p = q = 6$	Padé Approx. $p = q = 10$
-0.00010	-0.00010	-3.16992	0.91568	-0.00010	14370.508
-0.02500	-0.02500	-3.46994	0.82393	-0.02500	2166.5568
-0.02646	-0.02650	-3.54846	0.58361	-0.02848	1545.1549
-0.03780 ± 0.32935i	-0.03780 ± 0.32935i	-3.79015	0.36998	-0.03780 ± 0.32935i	1540.3456
-0.05475	-0.05475	-3.79481	0.29701	-0.05475	1445.2436
-0.06615	-0.06100 ± 0.32755i	-3.85081	0.10980	-0.06615	1434.9052
-0.08759 ± 0.10409i	-0.06615	-3.86392	-0.00327	-0.08759 ± 0.10409i	1019.4456
-0.11681	-0.08759 ± 0.10409i	-4.25558	-0.05199	-0.10759 ± 0.33539i	891.50938
-0.12665 ± 0.34150i	-0.11445 ± 0.78025i	-4.33068	-0.09677	-0.11681	795.91920
-0.13055 ± 0.17132i	-0.11681	-4.52052	-0.13551	-0.12906 ± 0.34552i	724.39851
-0.13922	-0.12818 ± 0.34639i	-4.68635	-0.15511	-0.13380 ± 0.17103i	648.25856
-0.13950	-0.13455 ± 0.17176i	-4.80909	-0.23989	-0.13417	625.18431
-0.13978	-0.17139	-4.84030	-0.32102	-0.17474 ± 0.27121i	593.37327
-0.14008	-0.17358 ± 0.27051i	-5.24457 ± 0.35652i	-0.34557	-0.17504	587.83144
-0.14027	-0.17504	-5.26514	-0.45854	-0.18411 ± 0.78161i	533.95381
-0.14048	-0.18208 ± 0.81259i	-5.67946 ± 0.81568i	-0.55539	-0.18562	528.11686
-0.14062	-0.18316 ± 0.81807i	-5.74580	-0.67482	-0.18892	519.93536
-0.14081	-0.18562	-5.80760	-0.73128	-0.20000	497.91600
-0.14104	-0.18877 ± 0.81637i	-5.98648	-0.95327	-0.20483 ± 0.87988i	456.93850
-0.14119	-0.18892	-6.10122	-0.97517	-0.20944 ± 0.36519i	420.89130

TABLE IV

COMPUTATIONAL BURDEN OF DIFFERENT METHODS TO COMPUTE EIGENVALUES USING GPU-BASED MAGMA LIBRARY

Model	Settings	Matrix setup	Matrix order	LEP sol.
No delays		1.18 s	$2,239$	11.91 s
Cheb. discr.	$N = 7$	29.4 s	$13,545$	12.69 m
Discr. of TIO	$N \cdot r = 21$	7.07 h	$40,635$	50.73 s
LMS approx.	$h = 0.2$ s	7.48 m	$32,895$	20.83 s
Padé approx.	$p = q = 6$	2.01 s	$3,415$	35.21 s
Padé approx.	$p = q = 10$	2.78 s	$4,399$	76.75 s

be expected to be the one based on Chebyshev discretization scheme. As indicated in [25], in fact, this approach shows a fast convergence. Moreover, simulation results on large scale systems indicate that the Chebyshev discretization does not require N to be high [32]. The accuracy of other methods can be thus defined based on a comparison with the results obtained through the Chebyshev discretization method. As shown in Table IV, the lightest computational burden is provided by Padé approximants. However, the solution obtained with $p = q = 6$ shows some differences with respect to the Chebyshev discretization, e.g., two poorly damped modes, namely $-0.26201 \pm 6.3415i$ and $-0.2746 \pm 5.9609i$ do not appear in the solution based on the Chebyshev discretization. Both modes show a damping lower than 5% and, through the analysis of participation factors [46], both are strongly associated with fictitious state variables introduced by the Padé approximant, e.g., (37) and (38). This effect has to be expected as, extraneous oscillations are a well-known drawback of Padé approximants. Finally, observe that for the Padé approximant with $p = q = 10$, results are fully unsatisfactory due to numerical issues. These are due to the extremely small values taken by coefficients in (36) and (38). For the considered case study, numerical problems show up for $p = q \geq 8$.

Tables III and IV also show that the techniques based on the TIO discretization and LMS approximation are both highly inaccurate and time consuming. In particular, the TIO discretization requires a huge time to setup up the matrix $\boldsymbol{S}_N$ of (22). It is likely that the implementation of the algorithm that build $\boldsymbol{S}_N$ can be improved using some sort of parallelization, which is not exploited in this case. However, the inaccuracy of the results makes unnecessary improving the implementation of this technique. Note also that the size of the computational burden of the LMS approximation strongly depends on the time step h used in (27). The smaller the time step h, the more precise the approximation, but the higher the computational burden. However, for $h < 0.2$, the size of the eigenvalue problem becomes too big and the MAGMA solver fails returning a memory error. Unfortunately, in this case, $h = 0.2$ s is too large to obtain precise results.

The results shown in Table III illustrate an extreme scenario with hundreds of highly stiff delays. To better understand the features of the four techniques discussed in the paper, it is useful to discuss two other scenarios, as follows:

- The all-island Irish transmission system with inclusion of only small delays, namely those associated with the AVRs of the conventional power plants.
- The all-island Irish transmission system with inclusion of only large delays, namely those associated with the reheaters of the turbines of conventional power plants.

Note that conventional power plants in the considered model of the all-island Irish transmission system are only 22, thus leading to 22 delays per scenario as opposed to the 296 delays of the all-delay scenario. Hence, the two scenarios above allow understanding the accuracy of the four techniques considered in the paper for a reduced number of delays with low stiffness. The 20 rightmost eigenvalues for the two scenarios above are shown in Tables V and VI, respectively. The main conclusions that can be drawn based on these tables are as follows.

- The consistency of the results of Tables III, V and VI confirms that the Chebyshev discretization is the most accurate and robust method of those considered in the paper. This has to be expected based on the discussion on the convergence properties of this method provided in [25] and references cited therein. Note also that there is no relevant difference between the three scenarios, i.e., all delays, small delays and large delays, with respect to the computational burden and approximation introduced by the Chebyshev discretization.
- Padé approximants work best for small delays and $p = q = 6$. They also provide consistent although not fully accurate results for large delays regardless the order of the approximants. The approximants may lead to numerical issues for small delays and high order (see the small-delay scenario with $p = q = 10$ in Table V). These issues are directly associated with the size of the LEP. The condition number of the state matrix, in fact, increases as the number of delays increases, as the values of the delays decrease and as the order of the Padé approximants increases.
- The TIO method is not particularly accurate regardless the number and the values of the delays. Results are more reliable for the small-delay scenario, although not all oscillatory modes are properly captured. According to the discussion provided in [25], to increase the points of the discretization grid, i.e., setting $N > 7$ would certainly increase the accuracy but, for large systems, the resulting matrix $\boldsymbol{S}_N$ becomes intractable.
- Finally, the LMS method appears to be the most inaccurate of all methods discussed in the paper for large LEPs. This method works slightly better for small eigenvalues, but results are poor in all scenarios considered in this case study (for example, no complex eigenvalue is found in the first 20 modes of the small-delay scenario in Table V). To increase the accuracy, one should decrease the time step h, but, as previously discussed, this option is not viable, due to memory constraints, for large problems like the one considered in this paper.

A rationale behind the poor results shown by the TIO and LMS methods compared to the Chebyshev discretization and Padé method, is as follows. The Chebyshev discretization method works directly with (14) and is concerned solely with approximating the $e^{-\lambda\tau_i}$ terms. As such, the quality of the results depends only on the quality of this approximation. In the same vein, Padé approximants are concerned with the approximation of $e^{-s\tau_i}$ terms in (32), and also in this case the quality of the approximation depends only the order of the approximants themselves. On the other hand, the TIO and LMS methods work with (12) and require approximations of both the $\dot{x}$ term and the delay terms. Hence, the TIO and LMS approximations require a step size small enough that the resulting difference equation is stable. Padé approximants and Chebyshev discretization are free of this step size restriction. This explains the particularly poor results with the LMS method, for which an explicit discretization is used and, consequently, needs a very small step size to achieve a stable approximation of the stiff DDAE. The RK based TIO method uses an implicit discretization, and so is somewhat better than the LMS method, as results provided in Tables V and VI indicate.

IV. Conclusions

This paper compares four different methods to approximate the spectrum of a DDAE system and applies such methods to a large-scale real-world power system. Among the proposed methods, the one based on Padé approximant shows the lightest computational burden but is less accurate than that based on Chebyshev discretization. The latter is the technique that provides the best ratio accuracy/computational burden. The other two techniques considered in this paper are both severely inaccurate and highly computationally expensive for large scale DDAEs.

The main conclusion of the paper is that the eigenvalue analysis of large state matrices as those obtained when considering DDAE models is feasible for real-world power systems provided that a sensible technique and numerical implementation are used. Future work will focus on determining the feasibility of algorithms that determine a subset of the spectrum of the DDAE, e.g., Arnoldi iteration, as well as on statistical analysis of power systems modelled as multi-delay DAEs.

Appendix

Chebyshev's Differentiation Matrix

Chebyshev's differentiation matrix $\boldsymbol{D}_N$ of dimensions $N + 1 \times N + 1$ is defined as follows. Firstly, one has to define $N+1$ Chebyshev's nodes, i.e., the interpolation points on the normalized interval $[-1, 1]$:

$$x_k = \cos\left(\frac{k\pi}{N}\right), \quad k = 0, \ldots, N. \tag{40}$$

Then, the element (i, j) differentiation matrix $\boldsymbol{D}_N$ indexed from 0 to N is defined as [47]:

$$\boldsymbol{D}_{(i,j)} = \begin{cases} \frac{c_i(-1)^{i+j}}{c_j(x_i - x_j)}, & i \neq j \\ -\frac{1}{2}\frac{x_i}{1-x_i^2}, & i = j \neq 1, N-1 \\ \frac{2N^2+1}{6}, & i = j = 0 \\ -\frac{2N^2+1}{6}, & i = j = N \end{cases} \tag{41}$$

where $c_0 = c_N = 2$ and $c_2 = \cdots = c_{N-1} = 1$. For example, $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ are:

$$\boldsymbol{D}_1 = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix}, \quad \text{with } x_0 = 1,\ x_1 = -1\ .$$

and

$$\boldsymbol{D}_2 = \begin{bmatrix} \frac{3}{2} & -2 & \frac{1}{2} \\ \frac{1}{2} & 0 & -\frac{1}{2} \\ -\frac{1}{2} & 2 & -\frac{3}{2} \end{bmatrix}, \text{ with } x_0 = 1,\ x_1 = 0,\ x_2 = -1\ .$$

TABLE V

20 RIGHTMOST EIGENVALUES FOR THE ALL-ISLAND IRISH SYSTEM – SMALL-DELAY SCENARIO

Chebyshev Discr. $N = 7$	Discr. of TIO $N \cdot r = 21$	LMS Approx. $h = 0.2$ s	Padé Approx. $p = q = 6$	Padé Approx. $p = q = 10$
-0.00010	-0.00010	-0.00026	-0.00010	109.19913
-0.02500	-0.00204	-0.00539	-0.02500	108.87505
-0.02646	-0.01022	-0.02693	-0.02641	108.61230
$-0.03780 \pm 0.32935i$	-0.02500	-0.06601	$-0.03780 \pm 0.32935i$	108.52138
-0.05475	-0.02606	-0.06974	-0.05475	108.25194
-0.06615	-0.04096	-0.10815	-0.06615	107.51873
-0.11681	-0.04804	-0.10833	-0.11681	106.81551
$-0.12643 \pm 0.34169i$	-0.05218	-0.10893	$-0.12638 \pm 0.34173i$	106.69809
$-0.13035 \pm 0.17151i$	-0.08719	-0.12668	$-0.13037 \pm 0.17149i$	106.50001
-0.13922	-0.09043	-0.13779	-0.13923	106.30230
-0.13950	-0.09740	-0.20357	-0.13948	106.29072
-0.13978	-0.10155	-0.23023	-0.13979	106.20415
-0.14008	-0.10413	-0.23879	-0.14008	105.75947
-0.14027	-0.11015	-0.24174	-0.14028	105.68017
-0.14048	$-0.11367 \pm 0.30866i$	-0.25728	-0.14047	105.53080
-0.14062	-0.12387	-0.26792	-0.14058	105.33107
-0.14081	-0.12449	-0.26812	-0.14081	104.46121
-0.14104	-0.13058	-0.27482	-0.14104	104.39888
-0.14119	-0.13158	-0.29058	-0.14118	103.87859
-0.14136	-0.13528	-0.32677	-0.14136	103.75343

TABLE VI

20 RIGHTMOST EIGENVALUES FOR THE ALL-ISLAND IRISH SYSTEM – LARGE-DELAY SCENARIO

Chebyshev Discr. $N = 7$	Discr. of TIO $N \cdot r = 21$	LMS Approx. $h = 0.2$ s	Padé Approx. $p = q = 6$	Padé Approx. $p = q = 10$
-0.00010	-0.05845	0.90982	-0.00010	-0.00010
$-0.01043 \pm 0.28380i$	-0.15447	$0.86571 \pm 0.80200i$	$-0.00768 \pm 0.31098i$	$-0.00768 \pm 0.31098i$
-0.02500	-0.19343	0.82424	-0.02500	-0.02500
-0.02644	-0.23237	0.81660	-0.02646	-0.02646
$-0.03780 \pm 0.32935i$	$-0.23840 \pm 0.22585i$	0.71504	$-0.03780 \pm 0.32935i$	$-0.03780 \pm 0.32935i$
-0.05475	-0.25508	$0.70500 \pm 1.11453i$	-0.05475	-0.05475
-0.06615	-0.25587	0.59069	-0.06615	-0.06615
$-0.08252 \pm 0.81721i$	-0.29481	0.58296	-0.11681	-0.11681
-0.11681	-0.31764	0.57749	$-0.12862 \pm 0.34003i$	$-0.12862 \pm 0.34003i$
$-0.12774 \pm 0.33959i$	-0.33064	0.54130	$-0.13257 \pm 0.17002i$	$-0.13257 \pm 0.17002i$
$-0.13305 \pm 0.17041i$	-0.34386	$0.52173 \pm 1.26923i$	-0.17473	$-0.16248 \pm 0.90009i$
$-0.17064 \pm 0.81911i$	-0.36627	0.42423	-0.17521	-0.17473
-0.17473	-0.36976	0.37890	$-0.17760 \pm 0.27345i$	-0.17521
$-0.17732 \pm 0.27363i$	-0.66239	$0.35163 \pm 1.35619i$	$-0.18222 \pm 0.91411i$	$-0.17760 \pm 0.27345i$
-0.17821	-0.67953	$0.28856 \pm 1.29241i$	-0.18551	-0.18551
$-0.18153 \pm 0.81508i$	-0.68946	$0.25343 \pm 1.17984i$	-0.18849	-0.18849
$-0.18387 \pm 0.37434i$	-0.70607	$0.23397 \pm 1.16515i$	$-0.19216 \pm 0.37807i$	$-0.19216 \pm 0.37807i$
$-0.18396 \pm 0.81173i$	$-0.74515 \pm 0.30043i$	$0.22110 \pm 1.39929i$	-0.20000	$-0.19959 \pm 6.25635i$
-0.18551	-0.75329	$0.14668 \pm 0.91392i$	$-0.21351 \pm 0.44644i$	-0.20000
-0.18849	$-0.78805 \pm 0.17382i$	$0.08521 \pm 1.33921i$	-0.21646	$-0.21351 \pm 0.44644i$

Acknowledgments

This work was conducted in the Electricity Research Centre, University College Dublin, Ireland, which is supported by the Electricity Research Centre's Industry Affiliates Programme (http://erc.ucd.ie/industry/). This material is based upon works supported by the Science Foundation Ireland, by funding Federico Milano, under Grant No. SFI/09/SRC/E1780. The opinions, findings and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the Science Foundation Ireland. The author has also benefit from the financial support of EC Marie Skłodowska-Curie Career Integration Grant No. PCIG14-GA-2013-630811.

References

[1] V. Venkatasubramanian, H. Schattler, and J. Zaborszky, "A Time-delay Differential-algebraic Phasor Formulation of the Large Power System Dynamics," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 6, London, England, May 1994, pp. 49–52.

[2] J. Nutaro and V. Protopopescu, "A new model of frequency delay in power systems," *IEEE Transactions on Circuits and Systems - II: Express Briefs*, vol. 59, no. 11, pp. 840–844, Nov. 2012.

[3] H. Wu and G. T. Heydt, "The Impact of Time Delay on Robust Control Design in Power Systems," in *Proceedings of the IEEE PES Winter Meeting*, vol. 2, Chicago, Illinois, 2002, pp. 1511–1516.

[4] H. Wu, K. S. Tsakalis, and G. T. Heydt, "Evaluation of Time Delay Effects to Wide-Area Power System Stabilizer Design," *IEEE Transactions on Power Systems*, vol. 19, no. 4, pp. 1935–1941, Nov. 2004.

[5] W. Yao, L. Jiang, Q. H. Wu, J. Y. Wen, and S. J. Cheng, "Delay-Dependent Stability Analysis of the Power System with a Wide-Area Damping Controller Embedded," *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 233–240, Feb. 2011.

[6] S. Wang, X. Meng, and T. Chen, "Wide-Area Control of Power Systems Through Delayed Network Communication," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 2, pp. 495–503, Mar. 2012.

[7] B. Yang and Y. Sun, "Damping Factor Based Delay Margin for Wide Area Signals in Power System Damping Control," *IEEE Transactions on Power Systems*, vol. 28, no. 3, pp. 3501–3502, Aug. 2013.

[8] M. Mokhtari, F. Aminifar, D. Nazarpour, and S. Golshannavaz, "Wide-area Power Oscillation Damping with a Fuzzy Controller Compensating the Continuous Communication Delays," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1997–2005, May 2013.

[9] W. Yao, L. Jiang, J. Wen, Q. H. Wu, and S. Cheng, "Wide-Area Damping Controller of FACTS Devices for Inter-Area Oscillations Considering Communication Time Delays," *IEEE Transactions on Power Systems*, vol. 29, no. 1, pp. 318–329, Jan. 2014.

[10] L. Cheng, G. Chen, W. Gao, F. Zhang, and G. Li, "Adaptive Time Delay Compensator (ATDC) Design for Wide-Area Power System Stabilizer," *IEEE Transactions on Smart Grid*, vol. 5, no. 6, pp. 2957–2966, Nov. 2014.

[11] C.-K. Zhang, L. Jiang, Q. H. Wu, Y. He, and M. Wu, "Delay-Dependent Robust Load Frequency Control for Time Delay Power Systems," *IEEE Transactions on Power Systems*, vol. 28, no. 3, pp. 2192–2201, Aug. 2013.

[12] A. Ali Pourmousavi and M. Hashem Nehrir, "Introducing Dynamic Demand Response in the LFC Model," *IEEE Transactions on Power Systems*, vol. 29, no. 4, pp. 1562–1572, Jul. 2014.

[13] H. Jia, N. Guangyu, S. T. Lee, and P. Zhang, "Study on the Impact of Time Delay to Power System Small Signal Stability," in *Proceedings of the IEEE MELECON*, Benalmádena, Spain, May 2006, pp. 1011–1014.

[14] H. Jia, X. Cao, X. Yu, and P. Zhang, "A Simple Approach to Determine Power System Delay Margin," in *Proceedings of the IEEE PES General Meeting*, Montreal, Quebec, 2007, pp. 1–7.

[15] S. Ayasun and C. O. Nwankpa, "Probability of Small-Signal Stability of Power Systems in the Presence of Communication Delays," in *International Conference on Electrical and Electronics Engineering (ELECO)*, vol. 1, Bursa, Turkey, 2009, pp. 70–74.

[16] S. Qiang, A. Haiyun, J. Hongjie, Y. Xiaodan, W. Chenshan, W. Wei, M. Zhiyu, Z. Yuan, Z. Jinli, and L. Peng, "An Improved Power System Stability Criterion with Multiple Time Delays," in *Proceedings of the IEEE PES General Meeting*, Calgary, Alberta, 2009, pp. 1–7.

[17] L. Ting, W. Min, H. Yong, and C. Weihua, "New Delay-dependent Steady State Stability Analysis for WAMS Assisted Power System," in *Proceedings of the 29th Chinese Control Conference*, Beijing, China, Jul. 2010, pp. 29–31.

[18] M. S. Mahmoud, *Robust Control and Filtering for Time-Delay Systems*. New York: Marcel Dekker, 2000.

[19] M. Wu, Y. He, and J. She, *Stability Analysis and Robust Control of Time-Delay Systems*. New York: Springer, 2010.

[20] A. Bellen and S. Maset, "Numerical Solution of Constant Coefficient Linear Delay Differential Equations as Abstract Cauchy Problems," *Numerische Mathematik*, vol. 84, pp. 351–374, 2000.

[21] A. Bellen and M. Zennaro, *Numerical Methods for Delay Differential Equations*. Oxford: Oxford Science Publications, 2003.

[22] D. Breda, S. Maset, and R. Vermiglio, "Pseudospectral Approximation of Eigenvalues of Derivative Operators with Non-local Boundary Conditions," *Applied Numerical Mathematics*, vol. 56, pp. 318–331, 2006.

[23] E. Jarlebring, K. Meerbergen, and W. Michiels, "A krylov method for the delay eigenvalue problem," *SIAM Journal on Scientific Computing*, vol. 32, no. 6, pp. 3278–3300, 2010.

[24] A. Butcher, H. Ma, E. Bueler, V. Averina, and Z. Szabo, "Stability of Linear Time-periodic Delay-differential Equations via Chebyshev Polynomials," *International Journal for Numerical Methods in Engineering*, vol. 59, pp. 895–922, 2004.

[25] D. Breda, "Solution Operator Approximations for Characteristic Roots of Delay Differential Equations," *Applied Numerical Mathematics*, vol. 56, pp. 305–317, 2006.

[26] E. Bueler, "Error Bounds for Approximate Eigenvalues of Periodic-Coefficient Linear Delay Differential equations," *SIAM Journal of Numerical Analysis*, vol. 45, pp. 2510–2536, Nov. 2007.

[27] K. Engelborghs and D. Roose, "Numerical Computation of Stability and Detection of Hopf Bifurcations of Steady-state Solutions of Delay Differential Equations," *Advances in Computational Mathematics*, vol. 10, no. 3-4, pp. 271–289, 1999.

[28] K. Engelborghs, T. Luzyanina, and D. Roose, "Numerical Bifurcation Analysis of Delay Differential Equations Using DDE-BIFTOOL," *ACM Transactions on Mathematical Software*, vol. 1, no. 1, pp. 1–21, 2000.

[29] K. Engelborghs and D. Roose, "On Stability of LMS Methods and Characteristc Roots of Delay Differential Equations," *SIAM Journal of Numerical Analysis*, vol. 40, no. 10, pp. 629–650, Aug. 2002.

[30] K. Engelborghs, T. Luzyanina, and G. Samaey, "DDE-BIFTOOL v. 2.00: a Matlab Package for Bifurcation Analysis of Delay Differential Equations," Department of Computer Science, K.U.Leuven, Leuven, Belgium, Tech. Rep., 2001, technical Report TW-330.

[31] G. A. Baker Jr. and P. Graves-Morris, *Padé Approximants - Part I: Basic Theory*. Reading, MA: Addison-Wesley, 1981.

[32] F. Milano and M. Anghel, "Impact of Time Delays on Power System Stability," *IEEE Transactions on Circuits and Systems - I: Regular Papers*, vol. 59, no. 4, pp. 889–900, Apr. 2012.

[33] V. Bokharaie, R. Sipahi, and F. Milano, "Small-Signal Stability Analysis of Delayed Power System Stabilizers," in *Procs. of the PSCC 2014*, Wrocław, Poland, Aug. 2014.

[34] E. Agullo, C. Augonnet, J. Dongarra, M. Faverge, H. Ltaief, S. Thibault, and S. Tomov, "QR Factorization on a Multicore Node Enhanced with Multiple GPU Accelerators," in *Proceedings of IPDPS 2011*, Anchorage, AK, Oct. 2010.

[35] M. Horton, S. Tomov, and J. Dongarra, "A Class of Hybrid LAPACK Algorithms for Multicore and GPU Architectures," in *Symposium for Application Accelerators in High Performance Computing (SAAHPC'11)*, Knoxville, TN, Jul. 2011.

[36] I. A. Hiskens, "Power System Modeling for Inverse Problems," *IEEE Transactions on Circuits and Systems - I: Regular Papers*, vol. 51, no. 3, pp. 539–551, Mar. 2004.

[37] U. M. Ascher and L. R. Petzold, "The Numerical Solution of Delay-Differential-Algebraic Equations of Retarded and Neutral Type," *SIAM Journal of Numerical Analysis*, vol. 32, no. 5, pp. 1635–1657, 1995.

[38] W. Zhu and L. R. Petzold, "Asymptotic Stability of Hessenberg Delay Differential-Algebraic Equations of Retarded or Neutral Type," *Applied Numerical Mathematics*, vol. 27, pp. 309–325, 1998.

[39] W. Michiels and S. Niculescu, *Stability and Stabilization of Time-Delay Systems*. Philadelphia: SIAM, 2007.

[40] K. Verheyden, T. Luzyanina, and D. Roose, "Efficient Computation of Characteristic Roots of Delay Differential Equations using LMS Methods," *Journal of Computational and Applied Mathematics*, vol. 214, pp. 209–226, 2008.

[41] P. M. Anderson and A. A. Fouad, *Power System Control and Stability*. New York, NY: Wiley-IEEE Press, 2002.

[42] J. M. Mauricio, A. Marano, A. Gómez-Expósito, and J. L. M. Ramos, “Frequency Regulation Contribution Through Variable-Speed Wind Energy Conversion Systems,” *IEEE Transactions on Power Systems*, vol. 24, no. 1, pp. 173–180, Feb 2009.

[43] P. Hirsch, *Extended Transient-Midterm Stability Program (ETMSP) Ver. 3.1: User’s Manual*, EPRI, TR-102004-V2R1, May 1994.

[44] J. L. Mathieu, S. Koch, and D. S. Callaway, “State Estimation and Control of Electric Loads to Manage Real-Time Energy Imbalance,” *IEEE Transactions on Power Systems*, vol. 28, no. 1, pp. 430–440, 2013.

[45] F. Milano, “A Python-based Software Tool for Power System Analysis,” in *Procs. of the IEEE PES General Meeting*, Vancouver, BC, Jul. 2013.

[46] P. W. Sauer and M. A. Pai, *Power System Dynamics and Stability*. Upper Saddle River, New Jersey: Prentice Hall, 1998.

[47] L. N. Trefethen, *Spectral Methods in Matlab*. Philadelphia: SIAM,

Federico Milano (S’02, M’04, SM’09) received from the Univ. of Genoa, Italy, the ME and Ph.D. in Electrical Eng. in 1999 and 2003, respectively. From 2001 to 2002 he was with the Univ. of Waterloo, Canada, as a Visiting Scholar. From 2003 to 2013, he was with the Univ. of Castilla-La Mancha, Spain. In 2013, he joined the Univ. College Dublin, Ireland, where he is currently an associate professor. His research interests include power system modelling, stability analysis and control.